



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Centre de la Imatge i la Tecnologia Multimèdia

# Creación del Arte de un Videojuego Pixel Art en Unity

## Treball Final de Grau Grau en Disseny i Desenvolupament de Videojocs

Cognoms: Solà Vila      Nom: Èric

Pla: 2014

Director: Ripoll, Marc

## Índex

Palabras Clave	4
Enlaces	4
Índice de Tablas	5
Índice de Figuras	5
Glosario	8
1. Introducción	9
1.1 Motivación	9
1.2 Formulación del problema	9
1.3 Objetivos generales del TFG	9
1.4 Objetivos específicos del TFG	9
1.5 Abasto del proyecto	9
2. Estado del arte	10
2.1 Técnicas de Animación	10
2.1.1 Animación Tradicional	10
2.1.2 Animación Esqueletal 2D	10
2.1.3 Rotoscopia	10
2.2 Pixel Art	11
2.3 Herramientas	12
2.1 Estudio de Mercado	13
3. Gestión del proyecto	14
3.1 Procedimiento	14
3.1.1 Cambios de Planificación	15
3.2 Herramientas de Comunicación	16
3.3 Herramientas de validación	16
3.4. DAFO	17
3.5. Riesgos y Plan de Contingencias	18
3.6. Análisis Inicial de los Costes	18
4. Metodología	19
4.1 Herramientas de Seguimiento	20
5. Desarrollo del Proyector	21
5.1. Aseprite y Unity	21
5.2. Animación: de 0 a “In Game”	23
5.2.1. Conceptualización	23
5.2.2. Creación	23

5.2.3. Animación	24
5.2.4. Sprite Sheet	26
5.2.5. Sprite Editor	27
5.2.6. Creación de Animaciones	28
5.2.7. Animator	29
5.3. Personajes	30
5.3.1. Aideen	30
5.3.2. Red Cap	31
5.3.3. Caorthannach	31
5.3.4. Banshee	32
5.3.5. Kelpie	32
5.3.6. Manannan Mac Lir	33
5.3.6.1. Portal	34
5.3.7. Morrigan	35
5.4. Objetos	37
5.5. Projectiles	37
5.6. Paleta de Colores	38
5.7. Tileset	38
5.8. UI	40
5.8.1. Botones	40
5.8.2. Textos	40
5.8.3. Corazones	41
5.8.4. Ilustraciones	41
5.8.5. Implementación en Unity	42
5.9. Partículas	43
5.9.1. Partículas Lanzar Fuego	44
5.9.2. Partículas Splash Sangre	45
5.10. SpriteSheets Finales	46
5.11. Limitaciones durante el Desarrollo	49
6. Bibliografía	50

## Palabras Clave

Pixel art, Animación 2D, Animación Tradicional, Videojuego, Mitología, Celtas, RogueLike

## Enlaces

[Repositorio del Proyecto.](#)

[Documento de Diseño](#)

[Aideen and the Dagda's Epiphany release\\_v1.1](#)

## Índex de taules

T1: DAFO.....	Pag. 17
T2: Riesgos y Contingencias.....	Pag. 18
T3: Costes del proyecto.....	Pag. 18

## Índex de figures

F1: Sprite Animación Esqueletal -----	Pag. 10
F2: Juego Enter the Gungeon-----	Pag. 11
F3: Juego Dead Cells -----	Pag. 11
F4: Diagrama de Gant -----	Pag. 14
F5: Interfaz Hack'n Plan -----	Pag. 20
F6: Tarea Hack'n Plan -----	Pag. 20
F7: Interfaz Aseprite -----	Pag. 21
F8: Interfaz Unity -----	Pag. 22
F9: Bocetos -----	Pag. 23
F10: Squash & Stretch-----	Pag. 24
F11: Anticipación de Disparo -----	Pag. 24
F12: Follow Through-----	Pag. 24
F13: Ease-in / Ease-out-----	Pag. 25
F14: Solidez-----	Pag. 25
F15: Sprite Sheet Aseprite -----	Pag. 26
F16: Animacion Separada Frame a Frame -----	Pag. 26
F17: Sprite Sheet Caothannach -----	Pag. 26
F18: Textura en Unity -----	Pag. 27
F19: Sprite Editor -----	Pag. 27
F20: Ventana Animación -----	Pag. 28
F21: Selección de Sprites -----	Pag. 28
F22: Animator -----	Pag. 29

F23: Animator con Transiciones	Pag. 29
F24: Transición Inspector	Pag. 29
F25: Referencias Celtas	Pag. 30
F26: Boceto Protagonista	Pag. 30
F27: Protagonista Final	Pag. 30
F28: Red Cap Original	Pag. 31
F29: Red Cap Final	Pag. 31
F30: Caorthannach	Pag. 31
F31: Banshee	Pag. 32
F32: Kelpie	Pag. 32
F33: Ataque Kelpie	Pag. 32
F34: Manannan mac Lir	Pag. 33
F35: Mac Lir Fase 1	Pag. 33
F36: Mac Lir Fase 2 y Triquetra	Pag. 33
F37: Mac Lir Fase 3	Pag. 33
F38: Animación de Tejido	Pag. 34
F39: Animación de Tejido 2	Pag. 34
F40: Portal	Pag. 34
F41: Morrigan	Pag. 35
F42: Morrigan Fase 1	Pag. 35
F43: Morrigan Transformación	Pag. 35
F44: Morrigan Forma Cuervo	Pag. 35
F45: Morrigan Fase 2	Pag. 36
F46: Morrigan Fuego	Pag. 36
F47: Morrigan Fase 3	Pag. 36
F48: Animación Vuelo	Pag. 36
F49: Vida y Triquetra	Pag. 37
F50: Orbe y Nudo de Roble	Pag. 37
F51: Proyectiles	Pag. 37
F52: Fuego Morrigan	Pag. 37

F53: Cambio Paleta	Pag. 38
F54: Paleta	Pag. 38
F55: Tiles Suelo	Pag. 38
F56: Tiles Pared	Pag. 39
F57: Tiles Esquina	Pag. 39
F58: Tiles Puerta	Pag. 39
F59: Barra Progreso	Pag. 40
F60: Botón	Pag. 40
F61: Título Principal	Pag. 40
F62: Insertar Texto	Pag. 40
F63: Corazones	Pag. 41
F64: Ilustraciones	Pag. 41
F65: Aideen delante de Fuego	Pag. 41
F66: Inspector del Botón	Pag. 42
F67: Inspector de Partículas	Pag. 43
F68: Inspector de Material	Pag. 43
F69: Inspector de Material 2	Pag. 43
F70: Parámetro Shape	Pag. 44
F71: Parámetro Color	Pag. 44
F72: Parámetro Shape 2	Pag. 45
F73: Parámetro Emission	Pag. 45
F74: Parámetro ParticleSystem	Pag. 45
F75: Parámetro Texture Sheet Animation	Pag. 45
F76: Tileset	Pag. 46
F77: Spritesheet Aideen	Pag. 46
F78: Spritesheet Jefes Finales	Pag. 47
F79: Spritesheet Objetos	Pag. 48
F80: Spritesheet Enemigos	Pag. 48
F81: Tiles Pared no Usadas	Pag. 49
F82: Animación Caída	Pag. 49

## Glossari

- **InGame:** Lo que está pasando mientras se está jugando a un videojuego.
- **Jugador Hardcore:** Jugador con gran dedicación e interés por los videojuegos.
- **Fotograma:** Cada una de las imágenes que componen una animación.
- **Procedural:** Que se genera aleatoriamente por el ordenador.
- **Runtime:** Se refiere a que se crea durante la ejecución del juego.
- **Ragdoll:** Animación generada por el motor de físicas que pretende imitar como actuaría un personaje inerte.
- **Hardware:** Partes físicas, circuitos y dispositivos que componen un sistema informático.
- **Indie:** En videojuegos se refiere a estudios independientes, sin muchos recursos.
- **Sprite:** Imagen 2D que se aplica a elementos dentro de un videojuego.
- **NPC:** Personaje que no está controlado por una persona.
- **Asset:** Cualquier elemento que compone un juego.
- **Glow:** Efecto visual que simula el brillo de la luz.
- **SpriteSheet:** Imagen 2D que contiene múltiples Sprites.
- **Tileset:** Imagen 2D que contiene una colección de Tiles.
- **Tile:** Imagen 2D que se utiliza como pieza para construir elementos más grandes dentro de un videojuego, como por ejemplo el escenario.
- **Timeline:** Es la línea de tiempo que nos indica la duración de una animación
- **Debug:** Es un término usado en el desarrollo para referirse a todo aquello que usan los desarrolladores para ver cómo está funcionando el juego.
- **Prefab:** Es un objeto complejo formado por una agrupación de objetos más simples de Unity.
- **Boss:** Personaje de un videojuego que acostumbra a ser más fuerte que los demás enemigos



# 1. Introducción

## 1.1 Motivación

El reto que supone para mí crear todo el arte de un videojuego es mi motivación principal para este proyecto.

Mi perfil siempre ha sido técnico, a día de hoy, me considero mucho más programador que artista pero cuando Nicolás Babot me informó sobre este proyecto no dudé en aceptarlo. En la carrera hemos dado asignaturas sobre animación 2D y 3D y como siempre me ha gustado la estética pixel-art así que pensé que sería una buena manera de aplicar todos esos conceptos en un proyecto real.

No sé si en el futuro volveré al pixel art o a la animación 2D pero creo que este proyecto es una gran oportunidad para aprender y sobretodo para estar orgulloso de haber creado un videojuego desde cero con un amigo.

## 1.2 Formulación del problema

En este caso no hay ningún problema a resolver sino más bien sería las ganas de aplicar todo lo aprendido en un videojuego.

## 1.3 Objetivos generales del TFG

El objetivo general consiste en crear un videojuego RogeLike desde cero ambientado en la mitología y el folklore celtas para la plataforma de PC.

## 1.4 Objetivos específicos del TFG

Este proyecto se realiza entre dos personas, mi compañero Nicolás Babot y yo.

Él se encarga de programar todo el juego mientras que yo me encargo de crear todos los gráficos. Así, mis objetivos específicos son:

- Diseñar y crear al personaje principal.
- Diseñar y crear los distintos enemigos y que encajen dentro de la mitología celta.
- Ambientar los distintos niveles dentro de la temática.
- Crear la interfaz del juego tanto in-game como en los distintos menús.

## 1.5 Abasto del proyecto

Las personas que se beneficien de este proyecto serán aquellas que deseen jugarlo, y aunque el target de este género de juegos suele ser mayoritariamente hardcore debido a la dificultad, pretendemos colgar el proyecto en distintas webs de forma gratuita para que todas las personas posibles puedan probarlo. A parte, también los desarrolladores nos beneficiamos porque, al fin y al cabo, es experiencia y curriculum para nosotros.

## 2. Estado del arte

Dentro de los videojuegos 2D encontramos distintos estilos a parte del pixel-art dependiendo de cada artista, de los recursos del proyecto o de las herramientas y técnicas que se utilicen.

### 2.1 Técnicas de Animación:

#### 2.1.1 Animación Tradicional:

Esta técnica se caracteriza por el hecho de crear fotograma a fotograma cada animación. Nace con el inicio de las imágenes en movimiento y sigue siendo una técnica muy importante sobretodo en videojuegos.

La ventaja principal de esta técnica es que ofrece al artista libertad creativa absoluta para todo aquello que se tenga que animar pero su inconveniente es que es la más lenta y requiere de cierta habilidad, no solo dibujando, sino también con los tiempos de las animaciones.

#### 2.1.2 Animación Esqueletal 2D:

Este tipo de animación nace del mundo de la animación 3D, donde los vértices de los polígonos se mueven influenciados por un esqueleto que está animado.

En 2D, este esqueleto afecta a las distintas partes de lo que queramos animar y para ello, al contrario que en animación tradicional, hay que dividir el sprite en distintos nodos que serán los que se verán afectados por el esqueleto como se ve en la Figura [F1].



F1 - Sprite Animación Esqueletal

Las ventajas que ofrece este tipo de animación es que una vez están todas las partes dibujadas hacer animaciones es muy rápido, se pueden aprovechar animaciones en personajes que sean anatómicamente parecidos, se pueden crear animaciones proceduralmente o crear animaciones en runtime como por ejemplo una animación de “ragdoll”.

Uno de los inconvenientes de esta técnica es el hecho de que estas sujeto al esqueleto y este te limita a la hora de poner al personaje en posturas que el rig no te permita.

Actualmente hay distintos softwares que permiten animar de este modo com Spine, Anima 2D, o Unity.

#### 2.1.3 Rotoscopia:

Es una técnica usada esporádicamente y seguramente la menos relevante de las 3 en el ámbito de los videojuegos. Consiste en dibujar fotograma a fotograma pero en este caso tenemos un video de la animación o distintas fotos que utilizaremos de plantilla.

Uno de los ejemplos más claros es el primer Prince of Persia donde el desarrollador grabó a su hermano haciendo todas las animaciones del personaje principal para posteriormente introducirlas en el videojuego.

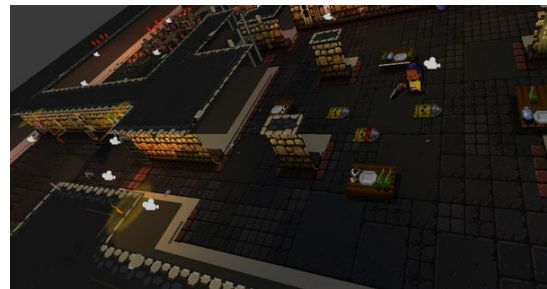
## 2.2. Pixel Art:

Antes de hablar sobre las aproximaciones modernas al pixel art hay que entender que esta estética nace de las limitaciones técnicas que tenían los ordenadores y consolas en los años 70-80. Lo que antes era una obligación debido al hardware ahora es una opción más a la hora de darle un estilo a tu juego. Desde su nacimiento y durante mucho tiempo predominó la animación tradicional aunque como veremos se ha ido modernizando.

En los últimos años este estilo se ha popularizado y distintos desarrolladores, sobretudo en el sector indie, han utilizado nuevas técnicas para darle un aspecto mejorado al pixel art. La mayoría de estas técnicas consiste en mezclar de algún modo elementos 3D con la temática 2D además de añadir elementos de postprocesado y de iluminación.

De este modo podríamos encontrar juegos como FEZ o Enter the Gungeon[F2] donde el escenario está en 3D y los personajes ya sean principales, enemigos o NPCs están dibujados en sprites 2D.

Por otro lado otra técnica que se ha popularizado consiste en crear los assets y animarlos en 3D y, además, renderizarlos a una resolución muy baja para que parezca que están dibujados. Los inconvenientes de esta metodología son que añade el trabajo de modelar en 3D y además hay que corregir las animaciones porque sino el renderizado a baja resolución genera mucho ruido. Pero una vez está modelado el personaje es más fácil sacar animaciones. Por ejemplo si haces un videojuego en vista cenital o 3/4 tendrías que hacer la misma animación desde distintas perspectivas mientras que de este modo haces una sola animación y la renderizas desde distintos puntos de vista.



F2 - Juego Enter the Gungeon



F3 - Juego Dead Cells

Lo que realmente le da al “nuevo” pixel-art una gran diferencia respecto al de antaño es el uso de la iluminación. Pudiendo lograr mejores ambientaciones y conseguir ese toque distintivo de realismo. Como se puede ver en la figura[dead cells], los efectos de las luces o el glow de los enemigos le dan un aspecto muy distinto a los del pixel-art tradicional.

## 2.3. Herramientas:

A la hora de hacer el arte de un videojuego se pueden utilizar distintas herramientas tanto para crearlo como para animarlo. Algunas de las más importantes dentro del 2D y el pixel art son:

- **Photoshop:** Es el editor de imagen por excelencia, multiusos, permite crear todo tipo de arte, aplicar filtros, tratamiento de imagen... A la hora de hacer animación tradicional es menos óptimo debido a que este no sería el uso para el que fue diseñado, de todos modos combina muy bien con herramientas de animación esquelética ya que permite hacer assets de gran calidad.
- **Spine:** Esta herramienta está centrada en animación esquelética 2D. Es fácil de usar, permite exportar fácilmente los SpriteSheets y tiene muchas características de softwares de animación 3D como editor de curvas para ajustar tiempos de animación, creación de meshes, huesos y skinning, además de Inverse Kinematics que permiten animaciones mucho más realistas. Tiene una versión de prueba, dos versiones de pago de 69\$ y 299\$ y una versión para empresas grandes de 2200\$ + 269\$ por usuario.
- **Aseprite:** Editor de imagen centrado exclusivamente en hacer animaciones pixel-art. Esta herramienta permite dibujar y animar tus sprites con animación tradicional, es fácil de usar y aprender. Además permite sacar automáticamente los SpriteSheets de tus animaciones. Tiene versión de pago de 14.99€ o también se puede conseguir de forma gratuita.
- **PixelEdit:** Esta herramienta permite dibujar y animar pero su punto fuerte es la capacidad para crear Tilesets. Dentro de su función de animar permite exportar SpriteSheets fácilmente. Dentro de su funcionalidad sobre los tilesets permite la exportación de los SpriteSheets y de información del tileset en formato XML o JSON. Tiene una versión de pago por 9.00\$ y una versión gratuita con limitaciones.

## 2.4 Estudio de Mercado

Al tratarse de un videojuego RogueLike hecho en pixel-art podemos encontrar numerosos productos parecidos bien por estética o bien por mecánicas de juego. Aunque este mercado está bastante saturado, ambos integrantes teníamos inquietudes en este tipo de juegos y como nuestro objetivo no es lucrativo decidimos seguir adelante.

### 2.4.1 Juegos del mismo género:

Los juegos RogueLike y RogueLite son aquellos juegos que tienen alguna especie de mazmorra o mapa generado proceduralmente y al morir, el jugador, debe empezar desde el principio. La diferencia entre ambos recae en la sensación de progreso permanente. A diferencia de los RogueLike, en los RogueLite el jugador durante las partidas se va desbloqueando elementos de juego que facilitaran partidas posteriores.

En el género RogueLite encontramos juegos como: Enter the Gungeon, Slay the Spire, The Binding of Isaac, Darkest Dungeon, One More Dungeon, Nuclear Throne, Rogue Legacy, Dead Cells...

En el género RogueLike encontramos juegos como: Spelunky o Downwell.

### 2.4.2 Juegos Gráficamente parecidos:

Aunque cada juego tiene su propio estilo y ambientación muchos de ellos se pueden englobar en la estética pixel art. Podemos encontrar gran variedad de títulos desde Owl Boy con unos sprites con mucha resolución y muy detallados hasta Undertale con un estilo más parecido a los juegos de los 80.

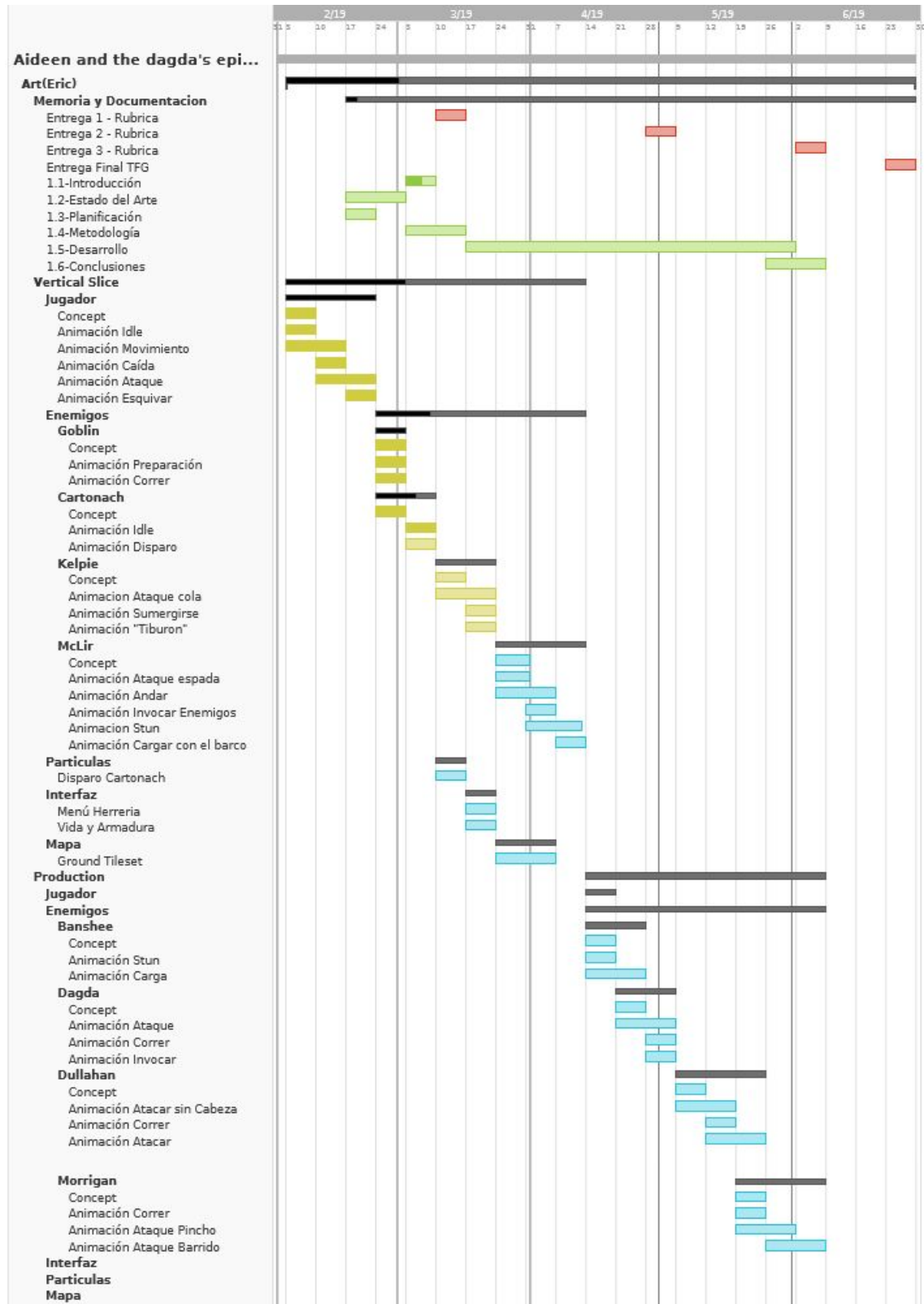
Esta estética se popularizó en los últimos años, sobretodo en la escena indie, debido a su coste relativamente bajo comparado con el 3D y la sensación de videojuego que transmite.

Algunos de estos juegos son: Mother Russia Bleeds, Enter the Gungeon, Celeste, Stardew Valley, Westerado, Bot Vice...

## 3. Gestión del proyecto

### 3.1. Procedimiento

En el siguiente Figura[F4] se puede observar como están las tareas distribuidas a lo largo del tiempo de vida del proyecto de manera general.



F4 - Diagrama de Gant

### 3.1.1. Cambios de Planificación:

Debido a un retraso y al cambio en la fecha de entrega del trabajo, se cambió la planificación de este y se amplió hasta el mes de setiembre. Aun así sobreestimé el tiempo que tendría durante el verano para trabajar en el proyecto.

Este error de sobreestimación de tiempo viene de que yo pensaba que podía trabajar al menos cinco horas cada tarde en el proyecto, pero debido a motivos personales esto se vio reducido a una hora total, de modo que algunas de las tareas que tenían menos prioridad fueron descartadas.

Algunas de estas tareas fueron tres enemigos: Dearg Due, Dagda y Dullahan. Por ejemplo Dullahan y Dagda tenían menos prioridad que Morrigan debido a que esta era la fase del boss final más espectacular de modo que los que eran más prescindibles quedaron en un segundo puesto. En cuanto al Dearg Due era el enemigo que menos encaja con el folklóre celta ya que, aunque forme parte de este, sus relatos y leyendas son muy posteriores a las de las demás criaturas.

Otra tarea que fue descartada fueron las tiles animadas del suelo así como las tiles de agua para los bosses Kelpie y Manannan Mac Lir. Esto último no fué tan grave ya que al final como la acción sucede en un bosque mágico en parte tiene justificación, pero de todos modos no debería haber sido así.

Además de esto, debido a que la planificación original era muy optimista con el tiempo durante el desarrollo del proyecto se cambió la idea del juego, se simplificaron algunos de sus aspectos y se quedó con la base con el fin de conseguir un juego lo más acabado posible, por encima del contenido total que tuviera.



### 3.2. Herramientas de Comunicación

Para coordinar el proyecto entre los 2 integrantes utilizamos distintas herramientas:

- **Whatsapp:** Para comunicación directa sobre problemas, dudas, horarios y distintos tipos de temas que surgen durante el día a día.
- **Discord:** Para nuestras reuniones durante la semana donde enseñamos lo que hemos estado haciendo, también para evaluar si algo funciona, ya sea una nueva mecánica implementada o una nueva animación, para buscar bugs e ir teniendo un seguimiento de estos y, finalmente al igual que whatsapp, para comunicación directa pero en este caso instantánea.
- **Google Drive:** Para compartir distintos tipos de archivos como el plan de costes, el gdd o documentación interesante que nos pueda servir para el desarrollo.
- **Github:** Esta es nuestra herramienta de control de versiones donde se encuentra el proyecto. En el siguiente enlace podéis ver como ha ido evolucionando y las distintas versiones del videojuego.

[Repositorio de Github](#)

### 3.3. Herramientas de Validación

Para la validación de las tareas, aprovechando que somos 2 integrantes, cada uno evalúa las tareas del otro.

A la hora de evaluar las tareas de programación miramos si lo que se está evaluando cumple con su propósito, después le comento la sensación que me transmite, si es buena o mala, y finalmente intento proponer maneras de cómo mejorarlo. A partir de aquí se sigue trabajando en ello hasta que está correcto y es jugablemente agradable.

A la hora de evaluar las tareas de arte, la primera fase es ver si se entiende el concepto y además, se adapta a la mitología celta. Posteriormente se escoge la mejor variación de este y se procede a animar. Cada animación sigue un proceso parecido y, en caso de que no pase el visto bueno de mi compañero, se sigue trabajando en base a los errores que él considere.

Se considera una tarea parcialmente acabada cuando arte y programación se juntan.  
Se considera completamente acabada cuando esa tarea ha pasado una fase de testeo y pulido final.

#### Validación de los bloques:

- Vertical Slice: Se centra en el combate con los enemigos y en conseguir una primera dungeon navegable. Esta primera fase se considera acabada si se cumplen los siguientes requisitos:
  - Movimiento y combate del jugador acabados.
  - Enemigo cuerpo a cuerpo y Enemigo a distancia acabados.
  - Generación procedural de un nivel.
  - Mini Boss y Boss parcialmente acabados.
  - Interfaz del juego acabada.
  - Tilesset del nivel 1 parcialmente acabado.

En caso de no cumplir estos requisitos para la fecha estimada se procederá a planificar el proyecto adaptándose al retraso acumulado.



- **Beta:** Se centra en conseguir una demo del juego que se pueda jugar de principio a fin aunque tenga fallos. Esta fase se considera acabada si cumple con los siguientes requisitos:
  - Herrería del jugador acabada.
  - Enemigo fantasma acabado.
  - Mini boss final y Boss final acabados.
  - Tileset del nivel 1 acabado y Tileset del nivel 2 parcialmente acabado.
  - Menú de Inicio acabado.

En caso de no cumplir estos requisitos para la fecha estimada se procederá a recortar contenido que no esté acabado para conseguir una versión lo más cerrada posible del juego.

- **Gold:** Es el juego final con los menores fallos posibles, en este caso nos centraremos en arreglar fallos y pulir el juego para que sea lo más divertido posible.

### 3.4. DAFO

Els punts forts i dèbils del tema i del seu desenvolupament

Fortalezas	Debilidades
<ul style="list-style-type: none"><li>- Tenemos experiencia usando Unity3D.</li><li>- Rejugable.</li><li>- Proyecto académico sin riesgo monetario.</li></ul>	<ul style="list-style-type: none"><li>- Poco contenido debido al tiempo.</li><li>- No hay nadie que tenga la capacidad de crear la música del juego.</li><li>- Al no centrarnos en el diseño puede que no sea divertido.</li><li>- Poco tiempo de desarrollo.</li></ul>
Oportunidades	Amenazas
<ul style="list-style-type: none"><li>- No hay prácticamente juegos con temática celta.</li></ul>	<ul style="list-style-type: none"><li>- Mercado saturado de juegos RogueLike.</li><li>- Mercado saturado de títulos indie.</li></ul>

T1 - DAFO

### 3.5. Riesgos y Plan de Contingencias

Riesgo	Solución
La música puede no estar acabada.	Ninguno de los dos tiene la capacidad de componer, la única solución sería buscar música online sin copyright o hacer una colaboración con algún músico.
Mala planificación del proyecto.	Puede que hayamos sobreestimado nuestras capacidades y no seamos capaces de conseguir todos los objetivos. En este caso la solución sería recortar contenido hasta conseguir un juego lo más cerrado posible.
Comunicación entre los integrantes	Es difícil que los dos integrantes del grupo tengamos la misma idea sobre lo que estamos desarrollando, hacia donde va el juego o como va a ser el resultado final de una mecánica o enemigo. La solución que tenemos planteada es hacer reuniones cada dos o tres días para ver cómo vamos en nuestras tareas y discutir sobre el desarrollo del juego.
Retraso Arte-Programación	Hay una clara diferencia de velocidad entre el desarrollo del arte y la programación lo que puede dar pie a que, en caso de retraso en alguna tarea, algunas cosas ya programadas tengan que retirarse del producto final. La solución más factible sería que, una vez llegados a la primera milestone, ver como ha sido, que tareas se han retrasado, como está el proyecto y en función de eso planificar de nuevo lo que está por delante.

## T2 - Riesgos y Contingencias

### 3.6. Análisis Inicial de los Costes

En la siguiente tabla se puede ver el coste de este proyecto durante 12 meses, en el siguiente enlace hay todos los datos. Al ser un proyecto no lucrativo aquí hemos decidido obviar los ingresos y las pérdidas que podría suponer. [Enlace Documento Presupuesto.](#)

Costes	1	2	3	4	5	6	7	8
Programador	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €
Artista	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €
Licencia Aseprite	15,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
Musica / FX	150,00 €	0,00 €	0,00 €	150,00 €	0,00 €	0,00 €	0,00 €	0,00 €
Unity	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
Ordenadores	43,61 €	43,61 €	43,61 €	43,61 €	43,61 €	43,61 €	43,61 €	43,61 €
Pantallas	4,17 €	4,17 €	4,17 €	4,17 €	4,17 €	4,17 €	4,17 €	4,17 €
Teclado	7,08 €	7,08 €	7,08 €	7,08 €	7,08 €	7,08 €	7,08 €	7,08 €
Total	4.119,86 €	3.954,86 €	3.954,86 €	4.104,86 €	3.954,86 €	3.954,86 €	3.954,86 €	3.954,86 €
Total Acumulado	4.119,86 €	8.074,72 €	12.029,58 €	16.134,44 €	20.089,31 €	24.044,17 €	27.999,03 €	31.953,89 €
	9	10	11	12				
	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €				
	1.950,00 €	1.950,00 €	1.950,00 €	1.950,00 €				
	0,00 €	0,00 €	0,00 €	0,00 €				
	0,00 €	0,00 €	0,00 €	0,00 €				
	0,00 €	0,00 €	0,00 €	0,00 €				
	43,61 €	43,61 €	43,61 €	43,61 €				
	4,17 €	4,17 €	4,17 €	4,17 €				
	7,08 €	7,08 €	7,08 €	7,08 €				
	3.954,86 €	3.954,86 €	3.954,86 €	3.954,86 €				
	35.908,75 €	39.863,61 €	43.818,47 €	47.773,33 €				

## T3 - Costes del proyecto

## 4. Metodologia

Nuestro proyecto utiliza la metodología preproducción - producción - postproducción y dentro de esta, para gestionar las tareas, utilizamos una estructura de metodología ágil.

**Preproducción:** En esta fase del proyecto creamos y discutimos el concepto del juego, la temática que tendría, el estilo artístico... en general es una fase en la que se le da forma a la idea inicial.

A parte de esto, los dos integrantes juntos, hicimos el GDD del proyecto donde se exponen los principales elementos del juego como:

- Movimiento y acciones del jugador.
- Estructura de la UI.
- Comportamiento de los enemigos.
- Tipo de generación de la mazmorra.
- Estilo Artístico.

Aunque el GDD nos marque la directriz a seguir está sujeto a cambios durante todo el proyecto. Además, para englobar todo esto dentro de la mitología celta, hubo una fase de documentación en la que investigamos sobre el folklore de los pueblos celtas.

**Producción:** En esta fase se desarrolla el juego en si, para ello, está dividida en distintas milestones con el fin de ir controlando como va evolucionando el proyecto a lo largo del tiempo. Estas son:

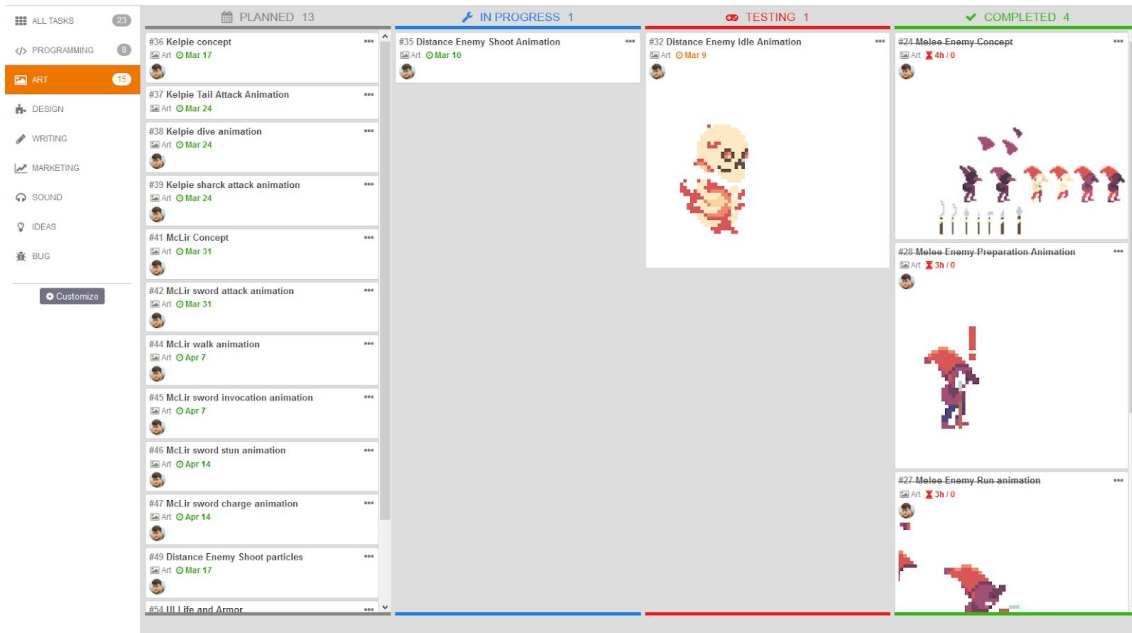
- **Vertical Slice:** Esta primera fase consiste en conseguir una demostración de los principales sistemas del juego para ver si funcionan y son divertidos. En nuestro caso el objetivo es conseguir un nivel completamente jugable.
- **Beta:** En esta fase del desarrollo el juego se considera acabado y no se añade nada nuevo. Debe ser jugable de inicio a fin y lo único que queda es arreglar fallos y pulir.

**Postproducción:** En esta fase del proyecto se testea el juego con el fin de conseguir arreglar el mayor número de fallos posible para conseguir la versión "Gold" del juego. Esta versión es básicamente el juego completamente acabado. Finalmente al acabar habría que escribir un "postmortem" del juego en el que se explica cómo ha sido el desarrollo, todo lo que ha ido bien, todo lo que ha ido mal...

Al ser un proyecto académico formado por 2 integrantes esta fase será mucho más pequeña que las dos anteriores debido al tiempo y a los recursos de los que disponemos

## 4.1 Herramientas de Seguimiento

Para evaluar cómo se está desarrollando el proyecto utilizamos la herramienta “Hack’n Plan” que nos permite saber qué tareas están por hacer, cuales se están haciendo actualmente y las que se han acabado, además de información adicional como el tiempo que se ha tardado o la fecha para la cual debe estar acabada esta tarea.



F5 - Interfaz Hack'n Plan



F6 - Tarea Hack'n Plan

En la figura[F5] se puede ver cómo está el progreso de la Vertical Slice, des de las tareas que quedan por hacer hasta las que están acabadas e implementadas dentro del juego.

En la figura[F6] se puede ver el nombre, el departamento, el tiempo que se ha tardado en realizarla y el resultado final. Esta es la información necesaria para llevar a cabo el seguimiento de cada tarea y el tiempo que se le dedica al proyecto.

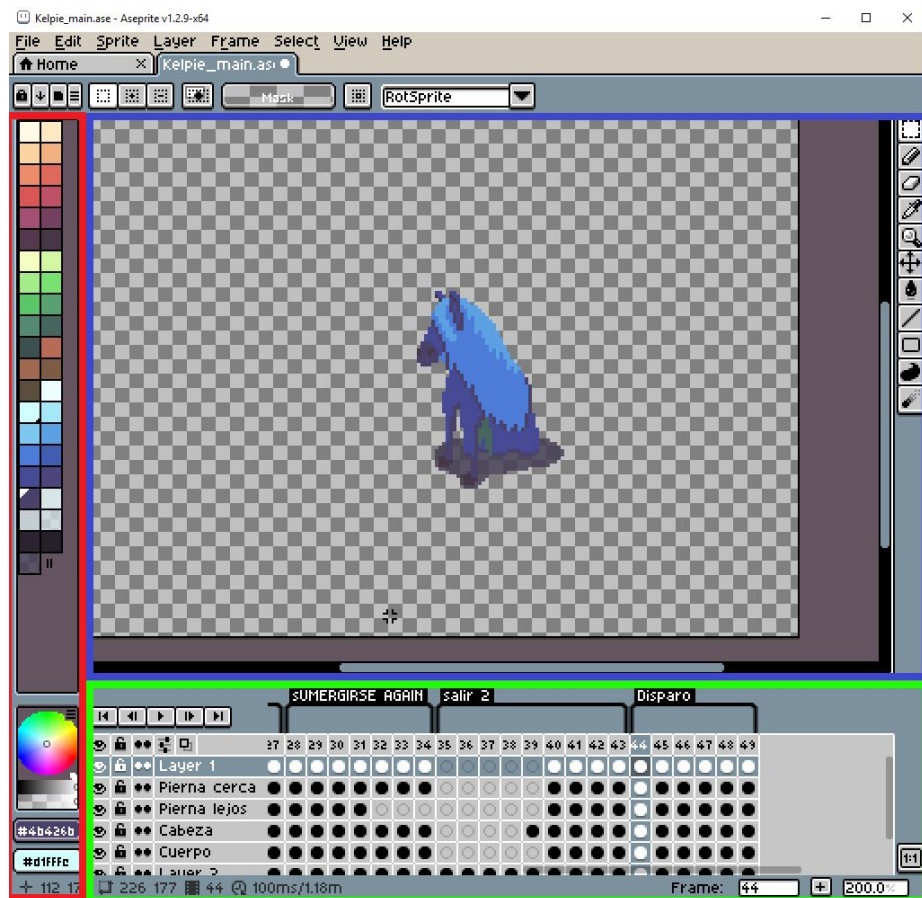
## 5. Desarrollo del Proyecto

### 5.1 Aseprite y Unity:

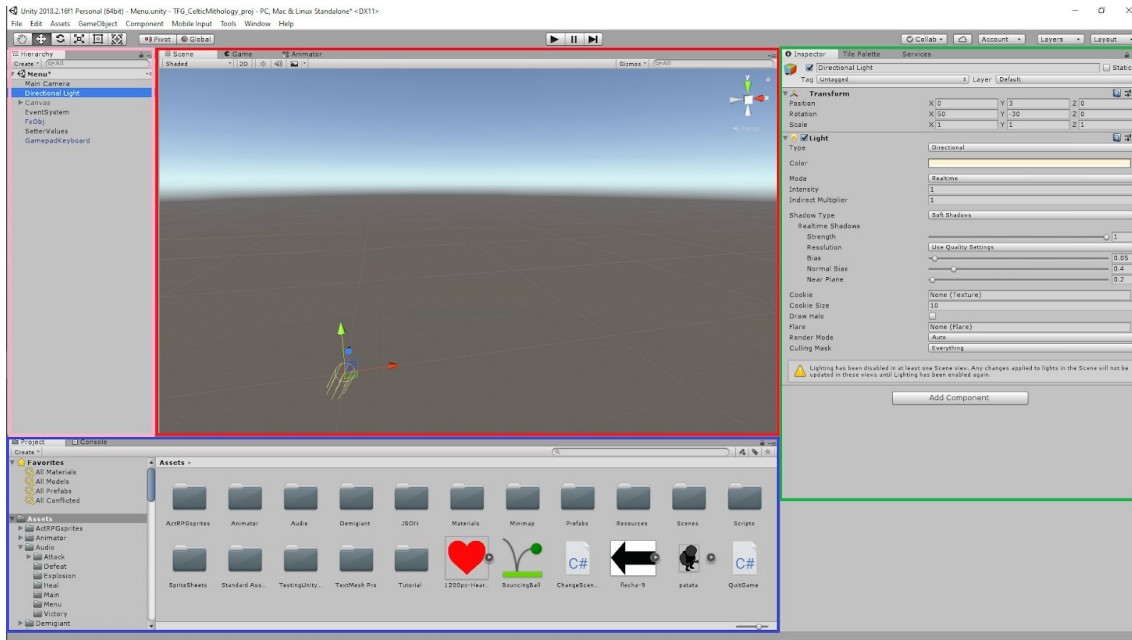
Para realizar todo el Pixel-art del juego he decidido usar Aseprite ya que se adapta perfectamente al proyecto, me permite crear y animar los distintos personajes así como los tilesets del entorno.

En la figura[F7] se puede ver como es la interfaz del programa y todo lo que nos ofrece.

- En la zona azul está el canvas del proyecto donde dibujaremos los assets y las distintas acciones que podemos hacer con el cursor.
- En la zona verde está todo lo relacionado con los frames y las distintas capas del dibujo. Aquí podremos modificar la velocidad de la animación, la opacidad y visibilidad de las capas, activar o desactivar el “papel cebolla” que nos permite ver frames anteriores o posteriores como guía a la hora de animar y, finalmente, la posibilidad de dar un nombre a cada animación para mantener el proyecto lo más limpio posible.
- En la Zona roja está la paleta de colores que estamos usando y todas las herramientas necesarias para trabajar el color.



F7 - Interfaz Aseprite



F8 - Interfaz Unity

En la figura[F8] podemos ver la interfaz de Unity.

- En la zona rosa tenemos la ventana de “hierarchy” que nos muestra todas las entidades que hay en nuestra escena, así como las relaciones que hay entre ellas.
- En la zona roja tenemos distintas ventanas:
  - Scene: nos muestra el entorno 3D en el que desarrollaremos el proyecto, nos permite navegar por él y además nos ofrece distintas opciones de debug para poder visualizar el comportamiento de entidades más fácilmente que luego en el juego final no se verán.
  - Game: nos muestra el punto de vista de la cámara de juego. Sirve para visualizar lo que vería el jugador, además de que nos sirve para ver mejor cómo quedaría la interfaz, ya que en la ventana de “scene” es mucho menos intuitivo.
  - Animator: en caso de tener una entidad seleccionada con un componente animator y un animator controller asociado a él, nos permite añadir animaciones a esta entidad y, en caso de estar testeando el juego, también nos permite ver que animación se está reproduciendo.
- En la zona verde tenemos la ventana de “inspector”. Esta ventana nos muestra información de aquello que tengamos seleccionado y la posibilidad de modificar distintos parámetros en función del tipo de objeto que sea. En este caso tenemos una entidad seleccionada con un componente de Luz y el inspector nos permite modificar propiedades de esta luz, la posición, rotación y escala de esta entidad y añadirle más componentes.
- En la zona azul tenemos la ventana de recursos. Nos permite visualizar y nos da acceso a todos los archivos del proyecto ya sean texturas, scripts, prefabs, escenas...

A parte de todo esto, para introducir todo nuestro arte en el juego, necesitaremos trabajar con otros aspectos de Unity que no se ven a simple vista como el SpriteEditor o la ventana de Animación. Más adelante explicaré como son y cómo se utilizan en profundidad.



## 5.2 Animación: de 0 a “In Game”:

Para desarrollar las todas animaciones de los distintos enemigos y personajes del juego he utilizado la misma metodología que está dividida en las siguientes fases:

### 5.2.1 Conceptualización:

En esta primera fase se analiza qué características debe tener el personaje que se va a desarrollar para ello se buscan referencias sobre cómo puede ser su apariencia o su actitud. Además, al estar ambientado en una temática, en nuestro caso el folklore celta, también se debe hacer un trabajo de documentación sobre su cultura, sus tradiciones, sus mitos y leyendas.

Por ejemplo, para el protagonista del juego, y siguiendo el documento de diseño que hicimos anterior al desarrollo del proyecto, este tenía que ser un herrero. Además, viendo que los pueblos celtas eran guerreros y siempre estaban peleando entre ellos decidí que un carácter simple y una apariencia muy varonil encajarían muy bien con el juego.

Una vez tenemos claro qué tipo de personaje queremos hacer pasariamos a la fase de creación.

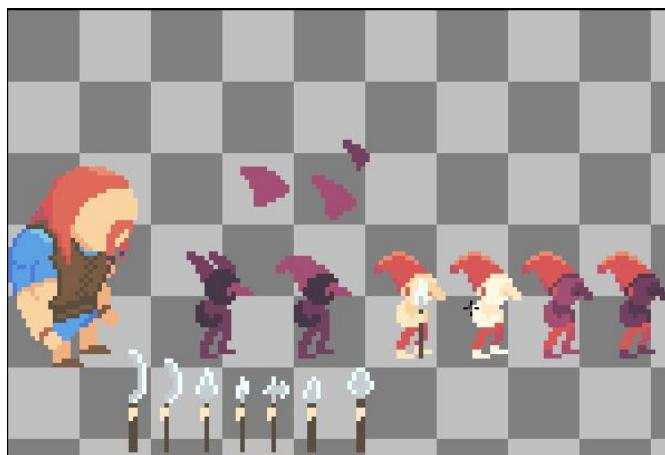
### 5.2.2 Creación:

El objetivo de esta fase es acabar con el personaje listo para empezar a animar, es decir, con su aspecto finalizado.

Siempre empiezo esta fase con unas siluetas del personaje y cuando una de ellas pasa el visto bueno se procede a colorear. En la figura[F8] se puede ver cómo sería este proceso. Primero hay las dos siluetas, una con sombrero y la otra sin. La silueta no debería tener distintos colores pero al trabajar con tan poca resolución me es mucho más sencillo ver de este modo como quedarían las partes que se superponen.

En este caso el duende con orejas de la figura[F9] quedó descartado debido a la leyenda en la que se basa este personaje y, probando con distintos sombreros, al final me quedé con el más largo porque me daba más juego a la hora de animar.

Posteriormente seguiría la fase de coloreado. Normalmente por el tiempo de desarrollo no se evalúa cómo quedaría el personaje con distintos colores pero, en este caso, aunque el duende de piel blanca se ceñía más al folklore celta podría no parecer un enemigo hostil, así que decidí que con colores más extraños favorece al gameplay.



F9 - Bocetos

### 5.2.3 Animación:

Esta es la penúltima fase del proceso en Aseprite y básicamente consiste en dibujar los distintos fotogramas de cada acción que deba realizar el personaje en el que se está trabajando. Al estar trabajando con Pixel-Art lo más común es utilizar la animación tradicional aunque para sprites muy grandes puede ser mejor animar el movimiento por partes.

En cuanto a principios de animación todos son importantes a la hora de aplicarlos en nuestro trabajo pero voy a hablar de los que personalmente tengo más en cuenta a la hora de animar.

- Squash & Stretch: Consiste en deformar el dibujo para aumentar la sensación de movimiento y así darle más dinamismo a la animación. En la figura[F10] se puede apreciar este efecto y, al aplicarlo, da la sensación de que se sumerge muy rápido.
- Anticipación: Consiste en hacer una preparación del movimiento antes de ejecutarlo. En videojuegos esto es muy importante de cara a los enemigos para que el jugador pueda reaccionar a tiempo a sus ataques, pero por otro lado hay que minimizarlo en el jugador porque puede que genere la sensación de pesadez o de que el juego responde con lentitud.



F10 - Squash & Stretch



F11 - Anticipación de Disparo

- Follow through: Este principio consiste en aplicar cierta inercia a las distintas partes que no efectúan el movimiento principal. En la Figura[F12] esta inercia se puede observar en la pequeña cresta que tiene el cuervo. Cuando este asciende la cresta tiende a quedarse abajo y cuando desciende todo lo contrario, de este modo da la sensación de que hay un pequeño desfase entre la cresta y el movimiento principal lo cual enriquece la animación.



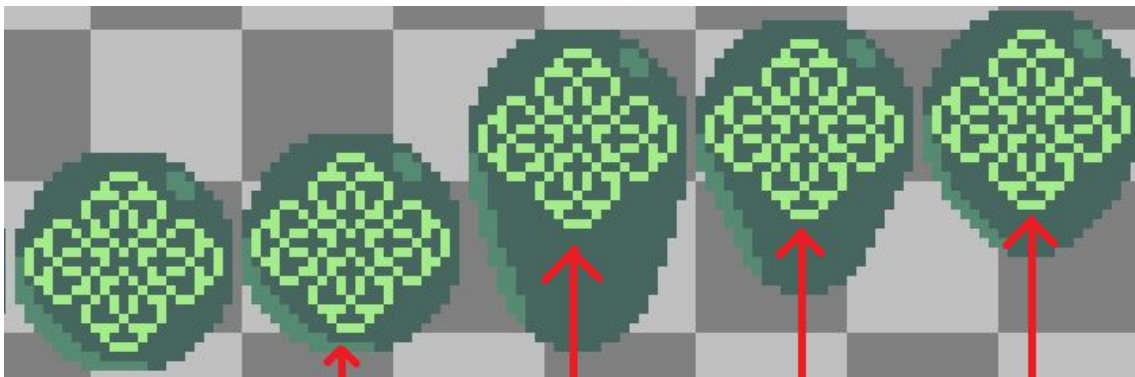
F12 - Follow Through



- **Ease in - Ease out:** Consiste en simular aceleraciones en nuestras animaciones de este modo parecen más orgánicas y naturales. En la figura[F13] se puede observar cómo este objeto tiene un fotograma al inicio en el que se mueve verticalmente muy poco (Ease-in) y luego acelera en el siguiente para frenar en los dos últimos (Ease-out). Estas aceleraciones se simulan haciendo movimientos más cortos o largos respecto al fotograma anterior.

El ease-in consiste en empezar despacio y acelerar, en este caso el orbe se mueve una unidad en el segundo fotograma respecto al primero y el tercero se mueve diez unidades respecto al segundo.

El ease-out es justo lo contrario, ir rápido y desacelerar despacio. En este caso el fotograma rápido es el tercero. El cuarto y quinto en lugar de frenar de golpe se desplazan una unidad respecto al anterior para dar esta sensación de desaceleración.



F13 - Ease-in / Ease-out

- **Solidez:** Este principio consiste en mantener la consistencia entre los distintos fotogramas y también en los dibujos. Sobre todo en Pixel-Art no obsesionarse con algún ángulo en el que quede raro el dibujo, hay fotogramas que individualmente no son perfectos debido a trabajar con bajas resoluciones pero mientras mantengas cierta coherencia entre los tamaños y formas, en conjunto, queda bien.

Para visualizar esto nos podemos fijar en el brazo derecho del personaje en la figura[F14]. Ninguno de los fotogramas es igual, ya sea en tamaño del antebrazo o del bíceps, por ejemplo en el primer fotograma el antebrazo es inexistente o en el tercer fotograma se ve un poco más pequeño que los demás brazos, pero al mantener todos cierta coherencia respecto a las formas del brazo, una vez está la animación en movimiento no se notan estas imperfecciones.



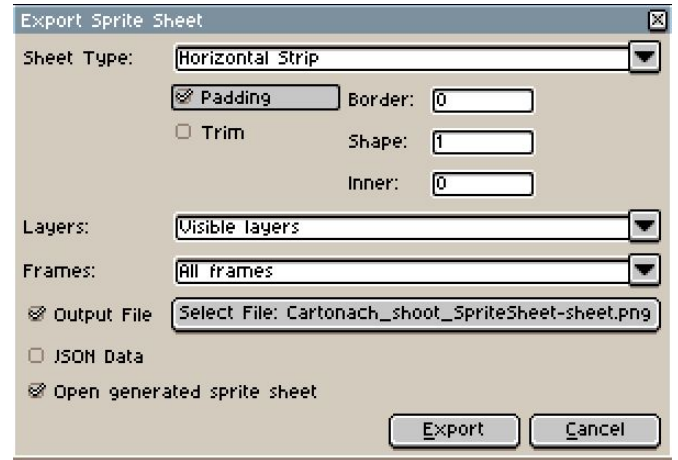
F14 - Solidez

### 5.2.4 Sprite Sheet:

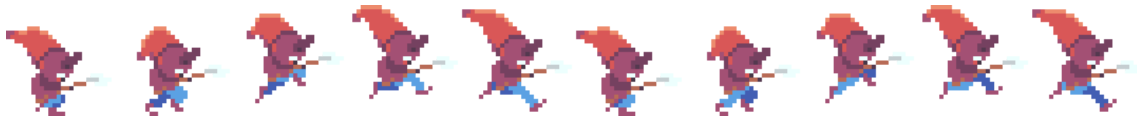
El sprite sheet es una imagen que aglutina todos los fotogramas de todas las animaciones de uno o más personajes.

Aseprite nos da una herramienta para exportar sprite sheets automáticamente. Esta herramienta es muy útil porque nos permite adaptar el sprite sheet a nuestras necesidades modificando ciertas opciones dentro del editor.

Personalmente me gusta trabajar con la opción de Horizontal Strip la cual nos dará los fotogramas ordenados de izquierda a derecha. Otra opción que tengo siempre activada es el “Padding” que sirve para separar los frames. Lo tengo establecido en 1 Pixel de distancia porque dentro de Unity si los fotogramas están pegados uno con otro, a veces, da algunos errores visuales.

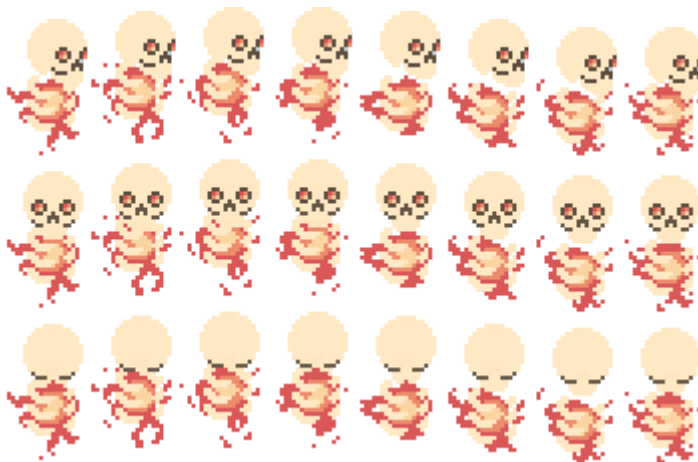


F15 - Sprite Sheets Aseprite



F16 - Animación Separada Frame a Frame

Una vez hemos sacado la Horizontal Strip de la animación nos quedaría un resultado como el de la figura[F16]. Ahora repetiremos el proceso para las demás animaciones y el siguiente paso será colocarlas en vertical dentro de nuestro sprite sheet respetando la altura ya que si se superponen nos dará errores en el siguiente proceso.



De este modo nos quedaría un resultado como el de la figura[F17] que ya estaría preparado para ser introducido dentro del motor gráfico.

F17 - Sprite Sheet Caorthannach

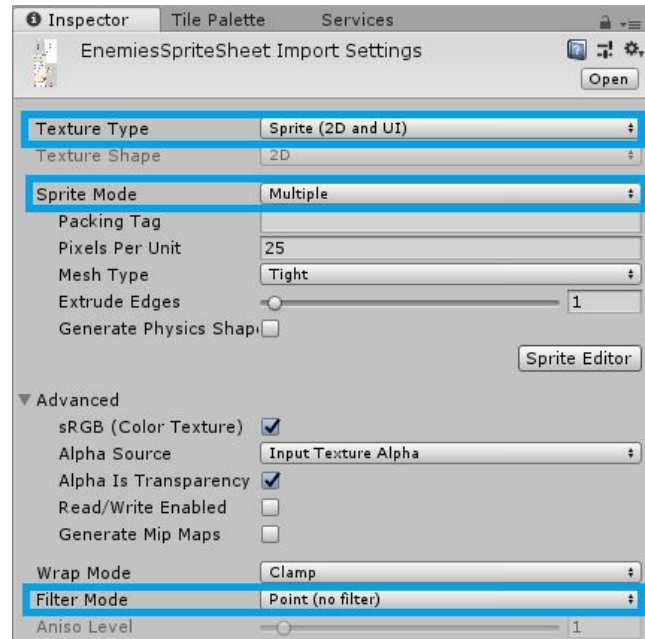
### 5.2.5 Sprite Editor:

Este es el primer paso dentro de Unity y consiste en dividir el sprite sheet en sprites individuales que podrán ser utilizados para crear las animaciones.

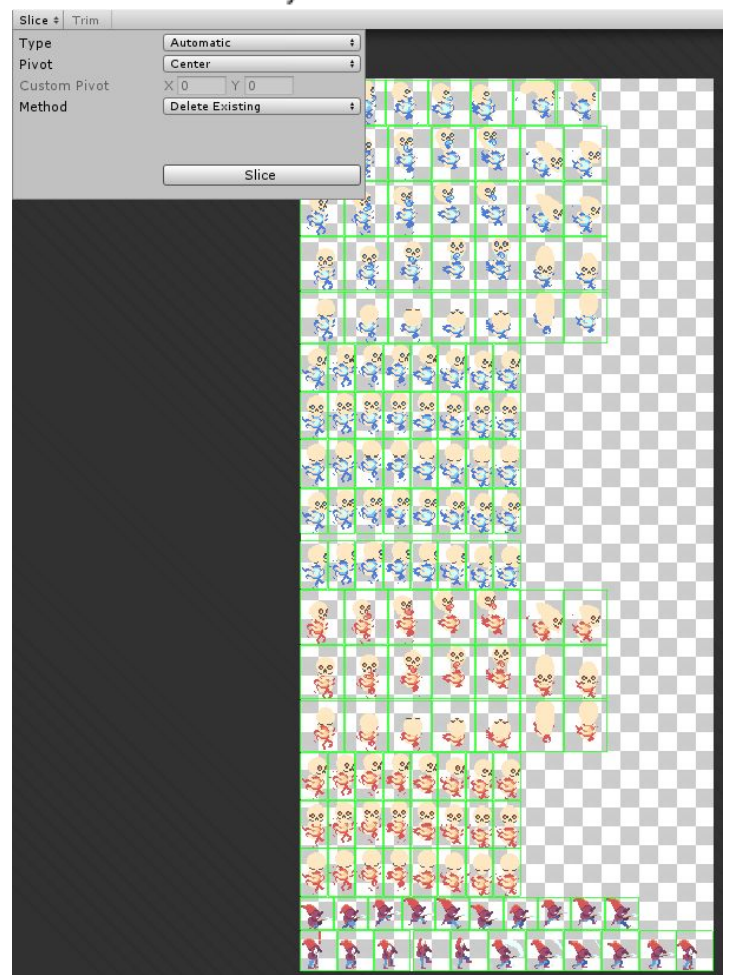
Para empezar hay que modificar el Texture type y elegir Sprite, cuando lo hagamos saldrán las siguientes opciones que se ven en la figura[F18]. Una vez hayamos hecho esto hay que cambiar el Sprite Mode a Multiple porque sino no podremos dividir el sprite sheet y, finalmente, modificar el Filter Mode a Point ya que sino la imagen se ve borrosa.

Una vez tenemos la textura lista para empezar a trabajar abriremos el Sprite Editor y empezamos a recortar los distintos fotogramas. Para ello Unity nos da distintas opciones, la primera es Automatic y como dice el nombre se encarga el motor de recortar el sprite sheet, las siguientes son por tamaño de celda y por número de celdas las cuales va bien si todos los sprites son del mismo tamaño y, finalmente, nos queda el modo manual haciendo click y arrastrando en el editor se generará un cuadrado y deberemos repetir el proceso para todos los sprites.

Una vez está todo el sprite recortado como en la figura [F19] seguimos con el siguiente paso, pero antes hay que saber que si se quiere modificar el sprite sheet y añadir más animaciones hay que hacerlo hacia arriba y hacia la derecha ya que sino los cortes que hayamos hecho previamente van a moverse y tendremos que volver a colocarlos en la posición correcta.



F18 - Textura en Unity

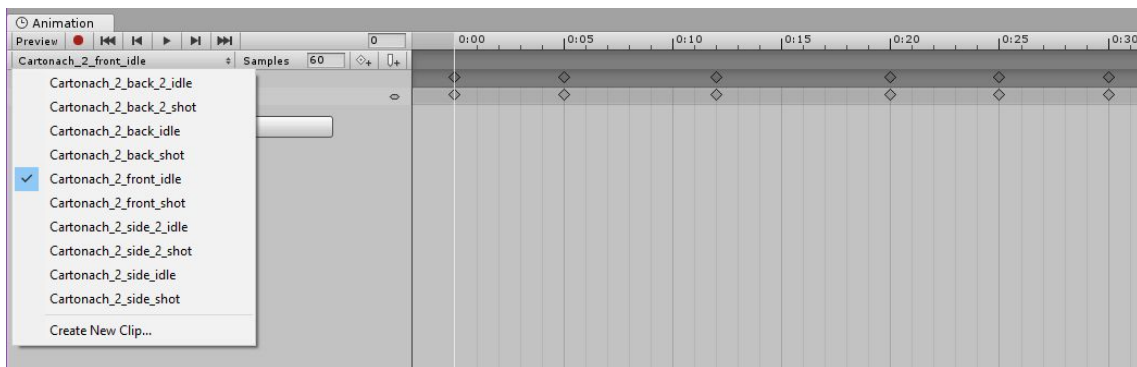


F19 - Sprite Sheet Editor

### 5.2.6 Creación de animaciones:

La manera en la que yo creo las animaciones en Unity es, inicialmente, añadiendo un componente Animator al GameObject que va a tener esas animaciones, creando un Animator Controller ya que el Animator lo necesita para funcionar y a partir de aquí, con el GameObject seleccionado, abriremos la ventana de Animación con “ctrl + 6” o “Window -> Animation -> Animation”

La ventana de animación se puede ver en la figura[F20] y su funcionamiento es muy sencillo. Primero crearemos una nueva animacion con “Create New Clip” y, como se observa en la figura[F21], abrimos el sprite sheet que previamente hemos recortado, seleccionaremos los sprites de la animación que estemos haciendo y los arrastraremos a la Timeline de la ventana de animación. Con esto la animación ya estaría hecha pero hay que modificar el tiempo que dura, para ello moveremos los frames en la timeline hasta que tengamos la animación con la duración adecuada.



F20 - Ventana de Animación



F21 - Selección de Sprites



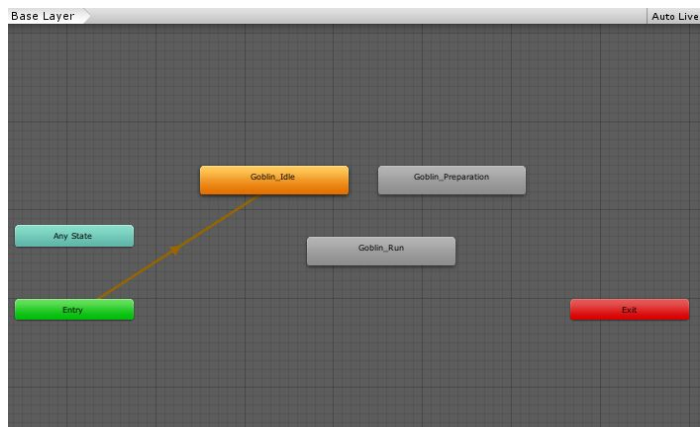
### 5.2.7 Animator:

Una vez tengamos las animaciones creadas y con el GameObject seleccionado abriremos la ventana Animator, debería verse como la figura[F22], en esta se puede observar las distintas animaciones pero ahora hay que añadir las transiciones entre ellas para que, des de código, se pueda cambiar de animación dependiendo de las acciones que haga el personaje.

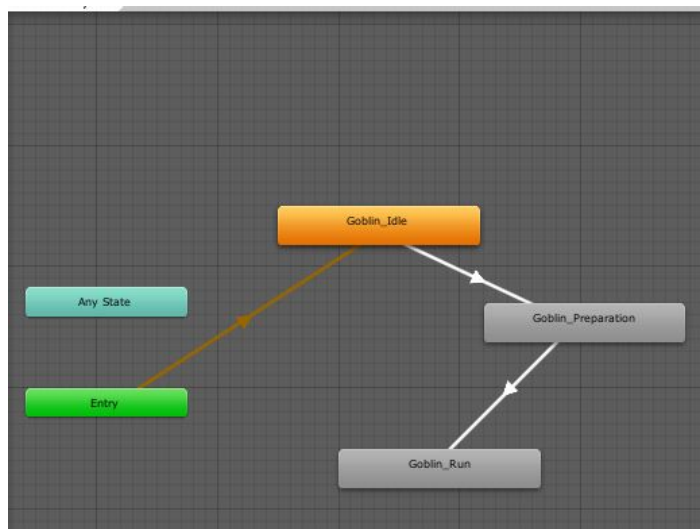
Para añadir una transición hacemos click derecho encima de un nodo, le damos “Make Transition” y seleccionamos la animación a la que puede transicionar. Después de este paso debería verse como en la figura[F23].

Las transiciones en pixel art es mejor que no tengan tiempos de transición entre una y otra ya que, sobretodo en el personaje principal, genera la sensación de que el juego no responde correctamente a las acciones del jugador. Al final cada transición debería verse como la figura[F20].

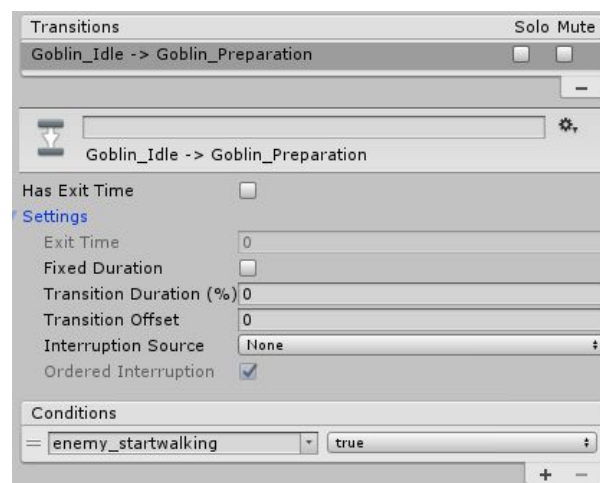
Finalmente, para que las transiciones se efectúen variando valores des de código, debemos añadir condiciones asociadas a una variable que pueda ser modificada. En la figura[F24] se puede ver como esta transicion se ejecutará cuando “enemy\_startwalking” sea true. Este último paso me encargo yo personalmente de hacerlo pero es normal que se haga conjuntamente entre programador y artista ya que el programador seguirá trabajando a partir de aquí para acabar de implementar las animaciones dentro del juego.



F22 - Ventana Animator



F23 - Animator con Transiciones



F24 - Transición en Unity

## 5.3 Personajes:

En este apartado voy a hablar sobre el proceso de creación de los distintos personajes y cómo encajan dentro del folklore/temática celta.

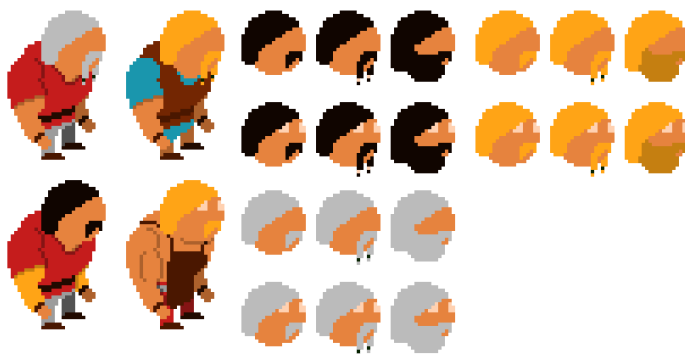
### 5.3.1 Aideen:

Cuando diseñamos el juego, una parte fundamental era la herrería y, aunque esto cambió a lo largo del proyecto, cuando empecé con el concepto del protagonista esto estaba muy presente. Algunas de las referencias que cogí se pueden ver en la siguiente figura[F25]:



F25 - Referencias Celtas

Como se trataba de un herrero la importancia de los brazos me pareció fundamental además que, al ser los celtas un pueblo guerrero y tosco, también creí que debía tener un aspecto muy masculino así que opté por un cuerpo de gorila con brazos y torso muy grandes y un tren inferior mucho más reducido. En la figura[F26] se puede ver la primera iteración del personaje y como acabó siendo finalmente en la figura[F27]. Del cambio en la paleta de colores hablaré posteriormente.



F26 - Boceto Protagonista



F27 - Protagonista Final

### 5.3.2 Red Cap:

Este es el enemigo más básico del juego, simplemente viene a por ti corriendo para atacarte. A la hora de buscar referencias de como hacer este enemigo tuve dificultades para encontrar algo que se ajustara al gameplay y también al folklore hasta que encontré al Red Cap. Esta criatura era un goblin agresivo, descrito como un señor pequeño con barba, armado con una lanza, un sombrero y ojos rojos, garras largas y una melena gris. La leyenda dice que su sombrero era rojo porque lo bañaba en la sangre de las personas a las que atacaba.

En la figura[F28] puede verse como es este goblin si nos ceñimos a la leyenda. Por otro lado si observamos el resultado final, figura[F29], podemos ver que he mantenido la lanza, el tamaño y el sombrero rojo, pero por otro lado no le he puesto pelo ni los ojos rojos, y el cambio más evidente: el color de la piel.

En cuanto al cabello y ojos, al ser tan pequeño y tener tan pocos píxeles con los que trabajar no quedaba demasiado bien, por otro lado el color de la piel fue una decisión tomada después de pasar la revisión. Debido a que con la piel blanca parecía de aspecto relativamente amigable le cambié el color hacia tonos morados para darle un aspecto de otro mundo, de algo desconocido.



F28 - Red Cap Original



F29 - Red Cap Final

### 5.3.3 Caorthannach:



F30 - Caorthannach

Para crear a un enemigo que disparase a distancia decidí utilizar la leyenda Caorthannach la escupe fuego. La leyenda dice que Caorthannach era un demonio con el que combatió San Patricio después de expulsar las serpientes de Irlanda. Cuando busqué referencias a la hora de decidir cómo sería me encontré que sus representaciones eran muy subjetivas, des de mujeres oscuras a mujeres de fuego, reptiles, demonios así que me tomé la libertad de adaptarlo más al gameplay.

En mi caso convertí a Caorthannach en un esqueleto de fuego, básicamente porque se trata de un enemigo estático que rota sobre si mismo para disparar en las distintas direcciones y el esqueleto me da la posibilidad de girar solo la cabeza hacia la dirección en la que tiene que disparar. Además también me permitia hacer una rápida conversión entre los dos tipos que había diseñados, uno que disparaba en cruz y otro que disparaba en cualquier dirección.

#### 5.3.4 Banshee:

Las Banshees son espíritus del folklore celta cuya aparición va ligada con el advenimiento de la defunción de una persona cercana. Para su representación hay distintos formas que pueden tomar pero me inspiré en la descripción de que es una mujer de aspecto fantasmal, vestida con un vestido blanco y una larga cabellera pelirroja.

Dentro del juego son un enemigo parecido al red cap pero con la diferencia de que solo es vulnerable una vez el jugador ha saltado por encima suyo. Para ello este personaje tiene dos animaciones



F31 - Banshee

#### 5.3.5 Kelpie:

Kelpie es el primer mini boss que tiene el juego. Según el folklore, los kelpie son unos espíritus de agua, capaces de cambiar de forma. Normalmente son descritos como caballos negros y poderosos que habitan cerca del agua, utilizan su capacidad de cambiar de forma para transformarse en personas e intentar engañar a humanos para atraerlos hacia el agua y una vez allí ahogarlos. La presencia de algas en el cabello es uno de los rasgos que permiten saber qué se trata de un kelpie y no de una persona real.

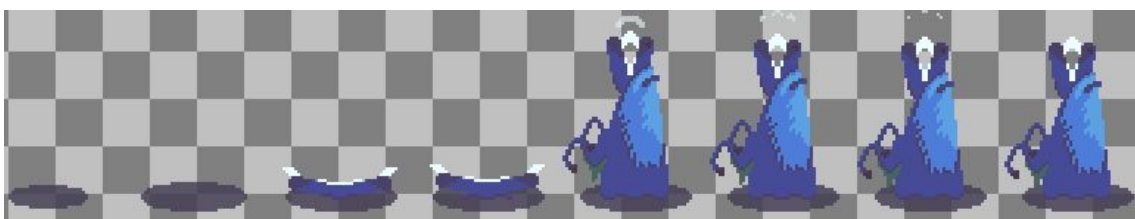
Sobre estas criaturas hay muchas representaciones en las cuales me he inspirado para crear al personaje, el único cambio que he hecho ha sido el color del pelaje, en lugar de usar negro puro lo he sustituido por azules y morados ya que está relacionado con el agua.

Esta actitud de atacar a los humanos y llevarlos hacia el agua para ahogarlos se ve reflejada en las animaciones y el diseño de sus ataques.



F32 - Kelpie

Kelpie persigue al jugador desde debajo del agua como si fuese un tiburón para luego salir y morderlo como se ve en la figura[F33] si falla el ataque quedará expuesto y podrá ser golpeado. Al ver que esto era no era suficiente para conseguir un gameplay entretenido decidimos añadir otro ataque en el que kelpie, al ser golpeado, dispara unos proyectiles de agua que rebotan por la sala dificultando así el movimiento del jugador. Este ataque es un ejemplo de algo que se sale completamente del folklore celta para favorecer al juego.



F33 - Ataque de Kelpie



### 5.3.6 Manannan mac Lir:

Manannan es el primer boss del juego. Según las leyendas celtas Manannan es una deidad acuática. Su papel era el de guiar las almas de los muertos hacia el otro mundo con su barco. Este personaje es descrito como un poderoso hechicero, posee una capa que es capaz de cambiar de color para mimetizarse con el entorno, la espada Fragarach capaz de traspasar cualquier armadura, una bolsa mágica, una coraza invulnerable y un casco mágico.

Como se puede ver en la figura[F34] en el diseño final ha prevalecido la capa y la espada. Todo lo demás al no verse traducido en acciones que haría dentro del juego preferí no añadirlo para mantenerlo más sencillo de animar y así ahorrar tiempo.

Mac Lir tiene tres fases en las que se refleja su leyenda. La primera consiste en atacar con su espada cuerpo a cuerpo como se ve en la figura[35]. Esta animación tiene una fase de anticipación bastante larga para que, como he dicho anteriormente, el jugador sea capaz de ver el ataque y reaccionar a tiempo. Como se puede ver, cinco de los ocho frames son de anticipación.

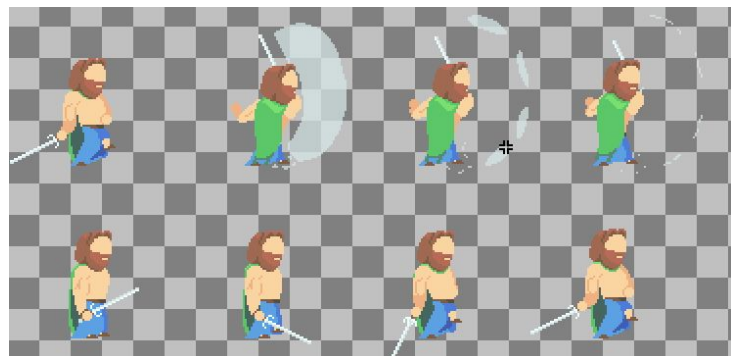
Su segunda fase consiste en invocar Banshees que, como he dicho anteriormente, son espíritus. Relacionándolo así con su faceta de guía de las almas al otro mundo. En esta fase, mientras dura la invocación, mac lir es invulnerable, esto lo reflejé con un círculo mágico a su alrededor como se ve en la figura[36], que lo pierde al acabar la invocación.

Este círculo tiene el simbolo de la Triquetra, figura[36] que en el folklore celta se utiliza para representar triadas, una de ellas: vida, muerte y renacimiento. Estando muy relacionada con Mac Lir ya que está a medio camino entre el mundo de los vivos y el de los muertos.

Finalmente en su última fase Mac Lir se sube a su barca fantasmal. En este momento el boss flota y la barca sale de debajo el suelo, además de que es semitransparente para demostrar que no es un barco normal como se ve en la figura[37]



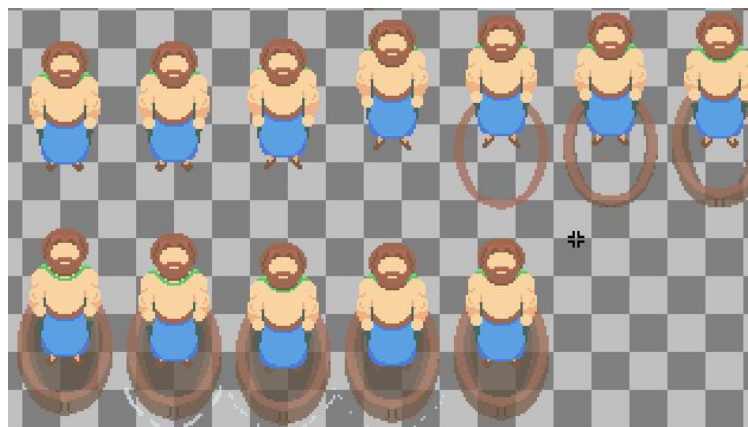
F34 - Manannan Mac Lir



F35 - Mac Lir Fase 1



F36 - Mac Lir Fase 2 y Triquetra

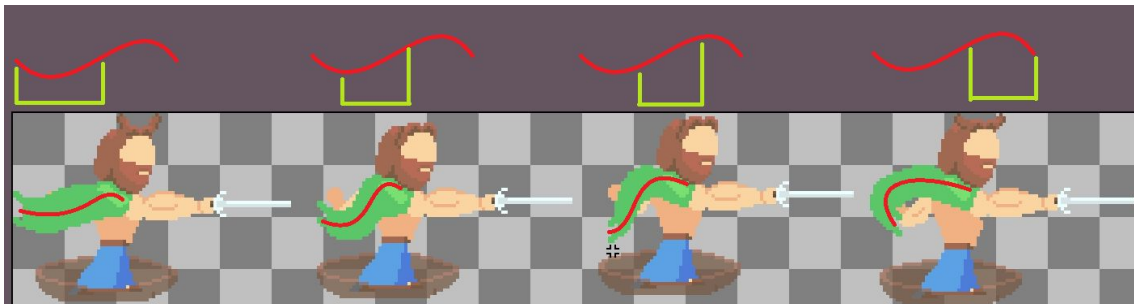


F37 - Mac Lir Fase 3

Posteriormente carga contra el jugador y en caso de fallar y chocar contra la pared su barco se destruye y queda vulnerable en el suelo. Para representar esto le salen unos patos en la cabeza para denotar que se ha dado un golpe y está aturdido. Como se puede ver en la figura[F78] al final del documento.

En cuanto al movimiento de la capa mientras carga creo que es un punto interesante dentro de la animación. En la figura[F38] tenemos la animación de carga en horizontal. Para animar una tela que se mueve por el viento hay que trazar formas siguiendo la oscilación de una onda.

En este caso se puede observar como cada capa forma una parte de la onda. Yo he utilizado una sola onda pero esto puede escalar en complejidad dependiendo de las ondulaciones con las que se dibuja el tejido.



F38 - Animación de Tejido

Lo mismo pasa con los distintos puntos de vista, como se ve en la figura[F39] tan solo hay que imaginar cómo sería esta ondulación vista desde otro ángulo.

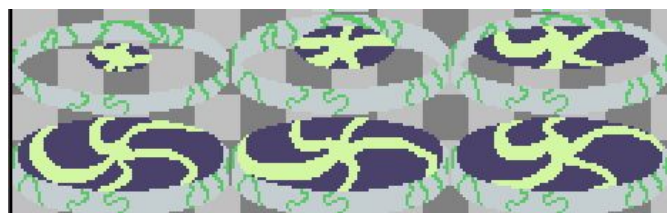


F39 - Animación de Tejido 2

#### 5.3.6.1 Portal:

Cuando Mac Lir muere en el centro de la sala aparece el portal que te lleva al siguiente nivel. Dicho portal está inspirado en un círculo de piedras estilo stonehenge pero a pequeña escala.

En la figura[40] se puede ver como es.



F40 - Portal

### 5.3.7 Morrigan:

Morrigan es el boss final del juego y está basada en la diosa celta del mismo nombre. Es la diosa de la guerra, la destrucción y la muerte.

Durante la guerra esta diosa se transformaba en cuervo y se encargaba de infundir valor a los soldados y temor a los enemigos en el campo de batalla. Dice la leyenda que si un soldado veía a Morrigan lavando su armadura significaba que iba a morir ese mismo día.

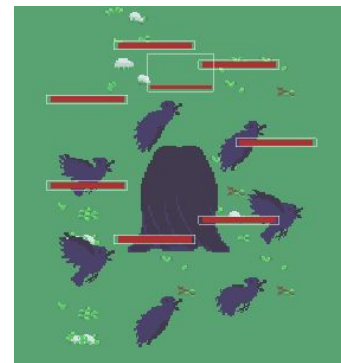
En cuanto a la representación de Morrigan, en el folklore llevaba armadura y una lanza y si no estaba transformada en cuervo solía tener muchos alrededor.

En mi caso la parte de la armadura y la lanza más propias de la diosa de la guerra han sido sustituidas por unas alas negras que creo que potencian su faceta de diosa de la muerte, además de que me ahorran trabajo a la hora de transformar a Morrigan.

En el juego Morrigan tiene tres fases. La primera muy ligada a su relación con los cuervos. Se cubre con sus alas y hace aparecer un montón de cuervos a su alrededor que atacarán al jugador y posteriormente ella se transformará y también le atacará. En la figura[F43] se puede ver el proceso de transformación y esta animación se hizo así para poder usarla en las dos direcciones de la transformación. Finalmente en la figura[F44] se puede ver a Morrigan en su forma cuervo.



F41 - Morrigan



F42 - Morrigan Fase 1



F43 - Morrigan Transformación



F44 - Morrigan Forma Cuervo

En su segunda fase Morrigan se cubre con las alas nuevamente pero esta vez también con fuego. Mientras eso ocurre invoca a unos esqueletos que sacaran sus manos del suelo e intentarán agarrar al jugador como se puede ver en la figura[F45], siguiendo así, con la temática de la muerte y la guerra. Después de esto la diosa lanza su fuego contra el protagonista véase la figura[F46].



F45 - Morrigan Fase 2



F46 - Morrigan Fuego

Finalmente en su última fase Morrigan invoca unos guerreros que perseguirán al jugador mientras ella le dispara bolas de fuego. En esta fase aparece el círculo mágico de Mac Lir pero en los colores de Morrigan para denotar que también está relacionada con la muerte como se ve en la figura[F47]



F47 - Morrigan Fase 3

Dentro de Morrigan encontramos la animación de volar tanto de ella en forma cuervo como de los cuervos pequeños, la de la figura[F48]. Para crear una animación de vuelo simple partimos del frame con las alas extendidas, en este caso el tercero, ya que este es el más bajo de todos.

Cuando bate las alas, como se ve en la figura[F48] hace un ease-out, sube varias unidades en el cuarto frame y en el quinto sube una unidad más para llegar a la altura máxima de la animación y, finalmente, el ave va abriendo las alas mientras desciende unidades hasta llegar al tercer frame de nuevo.



F48 - Animación Vuelo



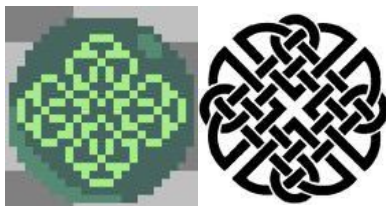
## 5.4 Objetos:

Dentro del juego tenemos dos elementos que el jugador puede coger: vida y fuerza.

- La vida es un objeto que los enemigos tienen una probabilidad de soltar al morir y está basado, al igual que el símbolo del círculo de Mac Lir, en la Triquetra con la diferencia de que esta está rodeada por un círculo como se puede ver en la figura[F49]. En la simbología celta el círculo representaba la unidad del espíritu y era una protección para que este no se rompiera por eso seleccioné este símbolo como curación dentro del juego.
- La fuerza por otro lado es un objeto que se encuentra al final de cada uno de los caminos de la mazmorra y al coleccionar tres el jugador puede subirse el daño o el número de corazones. Para representar este objeto utilicé el símbolo celta del nudo de roble, el cual era un árbol sagrado para los celtas y estaba asociado con el poder, la fuerza y el destino. El resultado final puede observarse en la figura[F50]



F49 - Vida y Triquetra



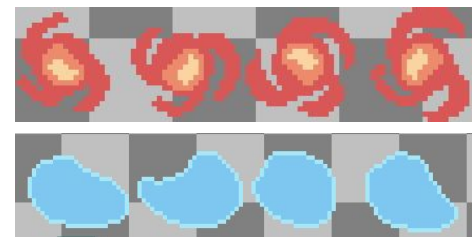
F50 - Orbe y Nudo de Roble

## 5.5 Projectiles:

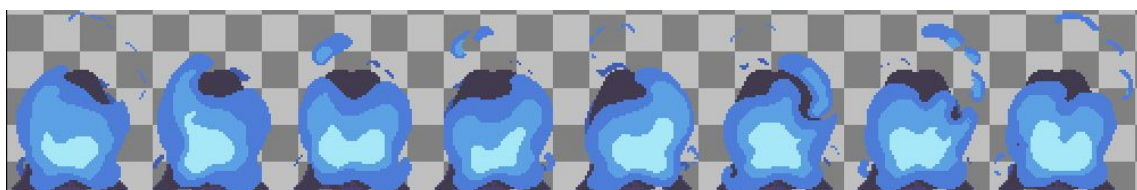
Hay dos proyectiles en el juego: las bolas de fuego de los caorthannach y Morrigan y las bolas de agua de Kelpie, figura[51].

Para crear las bolas de fuego y el fuego en general dentro del juego utilizo formas muy orgánicas y ondulantes. Empiezo con una forma base y frame a frame la voy moviendo según como creo que fluctuaría. A la hora de colorear el fuego hay que tener en cuenta que el interior es mucho más brillante y a medida que te acercas a las paredes se vuelve más oscuro.

Como se puede ver en la figura[F52]. Para las burbujas de kelpie utilicé la misma metodología que con el fuego, entendiendo que estos dos fluidos se mueven de maneras distintas. A la hora de colorear una burbuja es importante pintarla de un color y añadirle una pared que la recubra.



F51 - Proyectiles



F52 - Fuego Morrigan

## 5.6 Paleta de colores:

Cada juego tiene unas necesidades y la paleta de colores debe adaptarse a ellas. Cuando empezó el proyecto eso no lo tuve en cuenta y como resultado se puede ver en la figura[F53] como no hay ninguna armonía entre los colores ya que estaban escogidos al azar según mi criterio.

Para crear la nueva paleta figura[F54], utilicé un método que consiste en modificar los valores de Tono, Saturación y Luminosidad del color para darle mucha más riqueza visual.

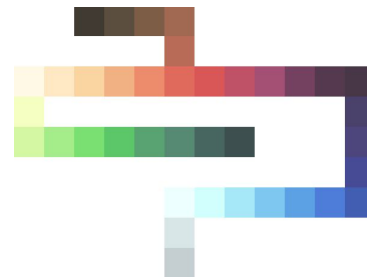
De este modo empiezas en un color cualquiera y para determinar uno más claro modificaremos el tono hacia uno más cálido, la saturación aumenta y la luminosidad también. Por lo contrario, si queremos crear un color más oscuro desplazamos el tono hacia un color más frío y reducimos la saturación y la luminosidad.

Por ejemplo para el rojo, si buscamos colores más claros iremos desplazandonos hacia el amarillo y si buscamos colores más oscuros hacia el morado.

Este método de generar los colores de la paleta no es perfecto pero nos permite crear una gama de colores muy rica sin tener muchos conocimientos sobre los colores.



F53 - Cambio de Paleta



F54 - Paleta

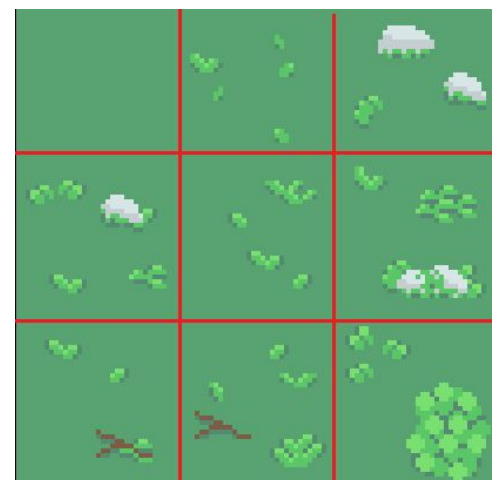
## 5.7 Tileset:

Toda la acción del juego se sitúa en un bosque mágico. Este está inspirado, por ejemplo, en bosques de zonas húmedas como galicia, irlanda, reino unido... en los que predominan los helechos y el musgo.

Debido al algoritmo del juego que generar el mapa aleatoriamente no se podían añadir muchos detalles ni variaciones en el suelo del bosque. Para solucionar este problema cree unas cuantas tiles, visibles en la figura[F55], que se distribuyen aleatoriamente por el suelo, teniendo prioridad la tile completamente verde para evitar generar demasiado ruido de fondo.

Todos los detalles de las tiles están hechos de la misma forma, tienen una parte más oscura que actúa como sombra, la base que sería el cuerpo del objeto y finalmente una parte más clara que simula un pequeño brillo.

Finalmente para crear una tile utilizamos la base verde y le añadimos unos cuantos elementos de los que hemos hecho previamente con distintos patrones para generar variedad.

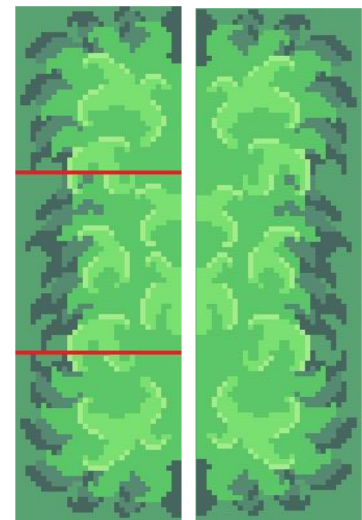


F55 - Tiles Suelo

En cuanto a las paredes de las salas opté por una zona frondosa de helechos que no dejase pasar al jugador. Debido a limitaciones del algoritmo, al igual que con el suelo, solo se pudo implementar un tipo de pared, ya que cada sala del mapa conoce de qué tipo son sus paredes pero no la de sus vecinos y, con más de un tipo de pared, la gran mayoría de veces no coincidían dos paredes adyacentes del mismo tipo y generaban fallos visuales graves.

En la figura[F56] se puede ver las tiles de pared. Para crear unas tiles de este tipo hay que tener en cuenta estas cosas:

- Los extremos de la tile central tienen que coincidir con la tile de finalización y con su extremo opuesto, de manera que se puedan colocar varias tiles centrales en caso de que la pared sea más larga de una tile. Como se ve en la figura[F56] las zonas rojas deben coincidir para que el resultado sea óptimo.
- Las tiles tienen que coincidir con ellas mismas simétricamente, de manera que al colocar la pared adyacente no se generen errores visuales y el resultado esperado sea como el de la figura[F56].



F56 - Tiles Pared

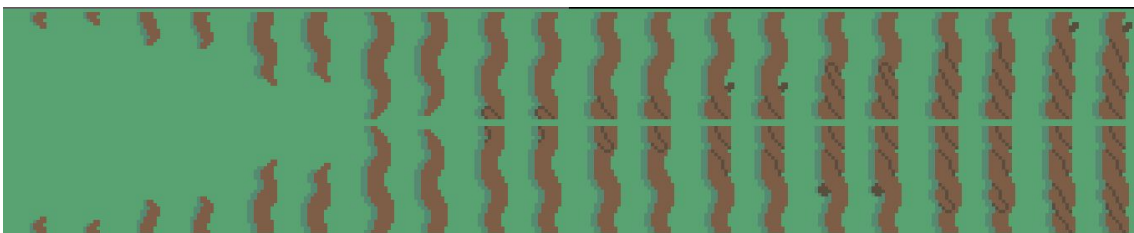
Por otro lado, el proceso para crear vegetación vista desde la parte superior es muy parecido al de los detalles del suelo. Empezamos con una base oscura y a medida que subimos niveles vamos cambiando los colores hacia tonos más claros, hasta que en el nivel superior añadimos algunos elementos que aporten riqueza a la tile. En mi caso añadí hojas de helechos para dar esa sensación de maleza.

Para las esquinas de las salas opté por hacer un tocón de árbol, tapado parcialmente por musgo denotando así el clima húmedo del lugar. El algoritmo nos dio el mismo problema que con las paredes, la imposibilidad de conocer la esquina de las salas adyacentes, de este modo solo se hizo un tipo de esquina formada por las cuatro tiles de la figura[F57].



F57 - Tiles Esquina

Finalmente para terminar con este apartado tenemos las puertas que unen las salas. Para reforzar el sentimiento de que es un bosque mágico las puertas que se abren y se cierran cuando el jugador entra y sale de las distintas salas son unas raíces que salen de bajo tierra y se entrelazan para impedir el paso como se puede ver en la figura[F58]



F58 - Tiles Puerta

## 5.8 UI:

A la interfaz del juego le he dado un estilo de roca antigua, cubierta por vegetación pero sin los relieves celtas ya que al tener muy poca definición generaba mucho ruido y, por ejemplo, en los botones dificultaba la lectura del texto. Este estilo se puede ver en la figura[F59] donde se muestra la barra de progreso.



F59 - Barra Progreso

### 5.8.1 Botones:

Para crear los botones del juego utilizaremos tres imágenes distintas, una para el botón en reposo, otra para cuando el usuario lo tenga seleccionado y, finalmente, una con el botón activado como se puede ver en la figura[F60].



F60 - Botón

Para dar la sensación de relieve al botón hay que oscurecer la parte inferior, en la figura[F60] resaltada con un círculo rojo, para que visualmente parezca una lado de este. Esta sería la imagen del botón en reposo.

El sprite del botón seleccionado es idéntico al de reposo pero se modifica la parte superior ya sea con un cambio de color o añadiendo algún efecto de manera que el usuario sepa en qué botón tiene el cursor, y si este puede ser pulsado o no. En este caso yo opté por un cambio de color hacia uno más claro.

Finalmente para crear la imagen del botón pulsado eliminamos la parte oscurecida en la de reposo y movemos la parte superior hacia abajo tantas unidades como tuviese la zona oscurecida, simulando así que el lateral del botón se ha escondido debajo.

### 5.8.2 Textos:

Otra parte de la interfaz son los títulos. Como se puede observar en la figura[F61] siguen el mismo estilo de roca antigua que los botones o la barra de progreso.



F61 - Título Principal

Para crear textos “pixelados” aseprite nos ofrece una herramienta muy útil en el menú de “Edit -> Insert Text” o con la tecla “T”. Se nos abrirá esta ventana que vemos en la figura[F62] en la que introduciremos el texto y podremos seleccionar el tamaño, la fuente y el color. Una vez tengamos el texto como se puede ver en la figura[F61] podemos añadir cualquier tipo de detalle, en mi caso le di el aspecto de roca antigua además de un pequeño relieve parecido al del botón pero en el lateral en lugar de la parte inferior.



F62 - Insertar Texto



### 5.8.3 Corazones:

Para representar la vida que tiene el personaje principal es muy común utilizar corazones como medidor. En un principio, mi objetivo era crear unos corazones con motivos celtas, es decir con las formas o con el clásico entrelazado que caracteriza la simbología de estos pueblos pero hacer esto en pixel art y con poca definición genera mucho ruido en la imagen y no queda bien así que al final opté por unos corazones clásicos, como se puede observar en la figura[F63].

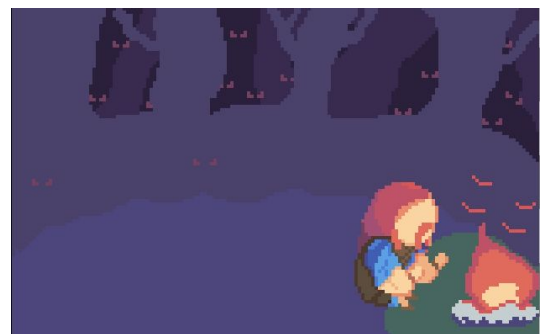


F63 - Corazones

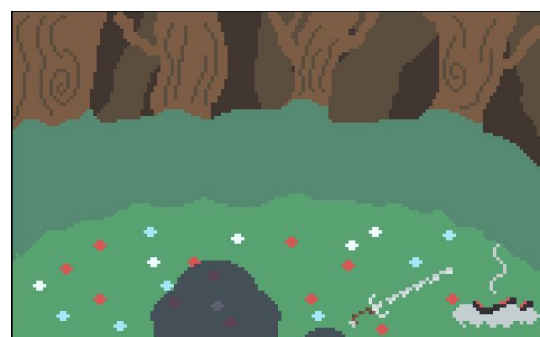
### 5.8.4 Ilustraciones:

Finalmente tenemos las ilustraciones que decoran los menús del juego. Son tres distintas: una para el menú principal, una para la escena de victoria y otra para la de derrota.

Como se ve en la figura[F64] la ilustración del menú principal representa a Aideen antes de la aventura, delante del fuego e ignorando todos los peligros que le rodean. Muy parecida a la de derrota, donde los peligros persisten pero Aideen ha perecido ante sus enemigos. Para crear una escena de noche como esta es mejor usar colores azules y morados oscuros antes que el negro puro, además de que favorecen a que todo aquello que es de este color forma parte de lo desconocido.



Finalmente tenemos la ilustración de victoria en la que toda la tonalidad de colores oscuros deja paso a los colores del bosque, además de que se puede observar la sombra de Aideen abandonando el lugar junto a otra sombra más pequeña.



Esto es un guiño a la que era la historia original del juego en la que Aideen perdía a su hijo e iba en su búsqueda, primero derrotando a Mac Lir para ver si todavía no se había llevado su alma y luego descubriendo que tenía que derrotar a Morrigan y Dagda para devolverlo a la vida.



F65 - Aideen delante del Fuego

Finalmente para dibujar a un personaje delante de un fuego trazo una línea imaginaria que simula el radio de iluminación del fuego y todo lo que está más alejado de esa línea lo oscurezco dos tonalidades de la paleta de colores. De este modo se crea este efecto de que una zona está muy iluminada y la otra no.

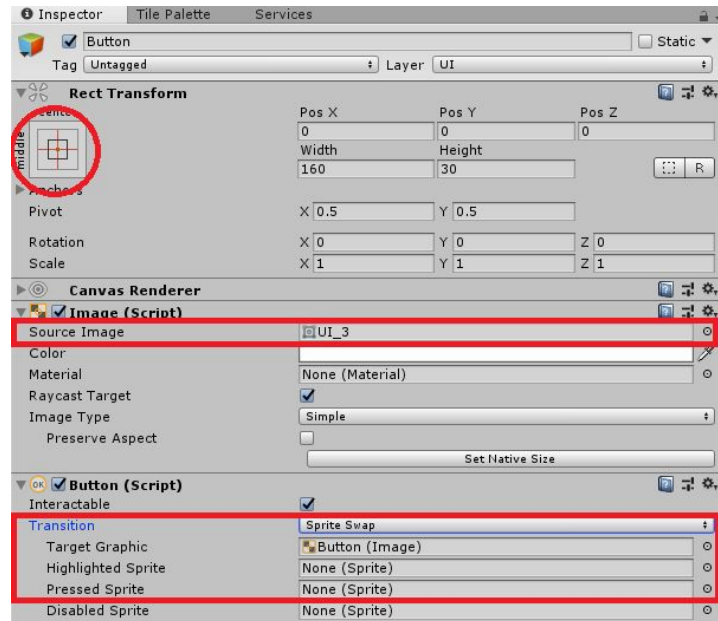
F64 - Ilustraciones

### 5.8.5 Implementación en Unity:

A la hora de implementar esto en Unity necesitaremos primero de todo un Canvas. Para ello en la ventana de “Hierarchy -> click derecho -> UI -> Canvas” para añadir un Canvas junto con un EventSystem que será el responsable de que, por ejemplo, los botones sean capaces de recibir las acciones del jugador.

Para implementar un botón:  
“seleccionamos el Canvas creado -> click derecho -> UI -> Button ” y se nos añadirá un botón como hijo del canvas. Si seleccionamos este botón y nos fijamos en la ventana de “Inspector” veremos algo parecido a la figura[F66].

En el componente “Image” y en apartado “Source Image” seleccionaremos nuestro sprite del botón en reposo y en el componente “Button” cambiaremos la opción de “Transition” a “Sprite Swap” tal y como está en la Figura[F66]. Finalmente en la opción de “Highlighted Sprite” seleccionaremos nuestra imagen de botón seleccionado y en la “Pressed Sprite” la de botón pulsado y con ello ya lo tendríamos implementado.



F66 - Inspector del Botón

Para implementar una imagen: “seleccionamos el Canvas creado -> click derecho -> UI -> Image” y se nos añadirá una imagen como hijo del canvas. Esta imagen tiene el mismo componente “Image” que el botón. Repetiremos el proceso de seleccionar nuestra imagen en el apartado de “Source Image” y ya estaría finalizado.

Una elemento importante a tener en cuenta cuando colocamos elementos de UI en la pantalla es el punto de ancla. Dependiendo de donde coloquemos el elemento hay que modificar el punto de ancla para que, cuando rescalemos el tamaño de la ventana los elementos sigan en la posición que deben. Como se ve en la figura[F66] para cambiar el punto de ancla hay que seleccionar la zona dentro del círculo rojo. Por ejemplo si un elemento va colocado arriba a la izquierda de nuestra pantalla el punto de ancla debe ser la esquina superior izquierda.

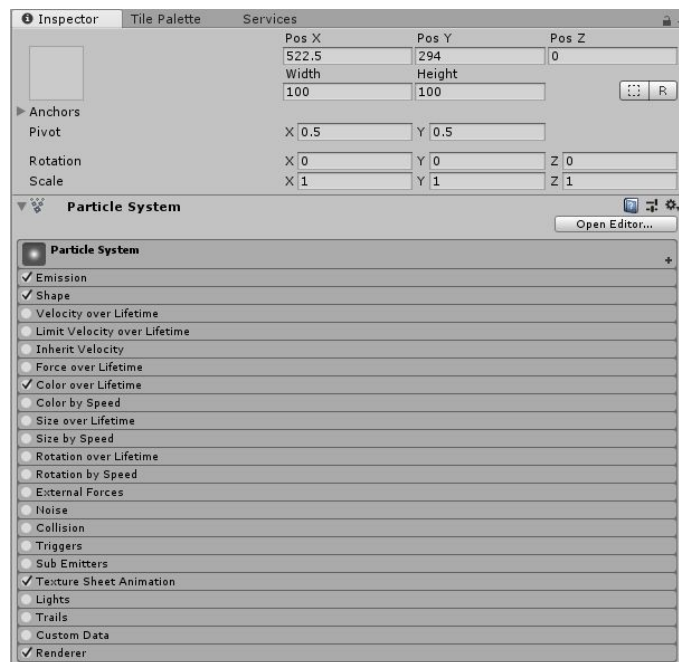
## 5.9 Partículas:

Dentro del juego hay dos tipos de partículas: El fuego de Morigan y la sangre de los enemigos.

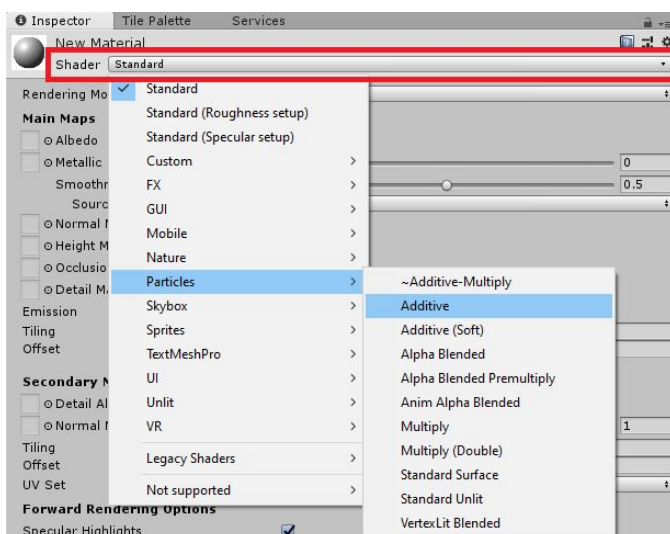
Para implementar las partículas dentro del juego es necesario crear un "ParticleSystem", para ello en la ventana de "Hierarchy -> click derecho -> effects -> ParticleSystem". En la ventana "Inspector" del "ParticleSystem" veremos lo mismo que en la figura[F67].

Además necesitaremos un material para las partículas. Para ello en la ventana de "Recursos -> click derecho -> Create -> material". Con el material seleccionado, en la ventana de inspector, debemos cambiar el Shader tal y como se muestra en la figura[F68]. Para añadir el material al "ParticleSystem" hay que arrastrarlo de la ventana de recursos hasta la entidad de la ventana "Hierarchy" correspondiente.

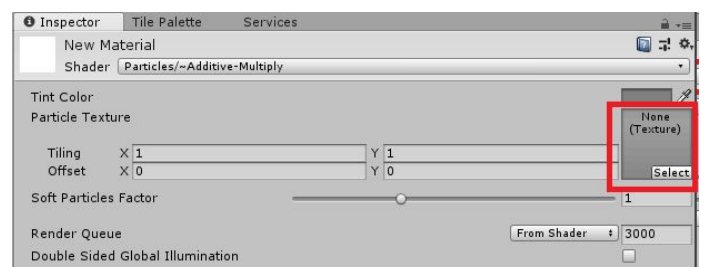
Una vez hemos cambiado el Shader del material tan solo hay que asignar nuestro sprite como textura tal y como se muestra en la figura[F69]. De este modo ya podemos empezar a explorar el "ParticleSystem".



F67 - Inspector de Partículas



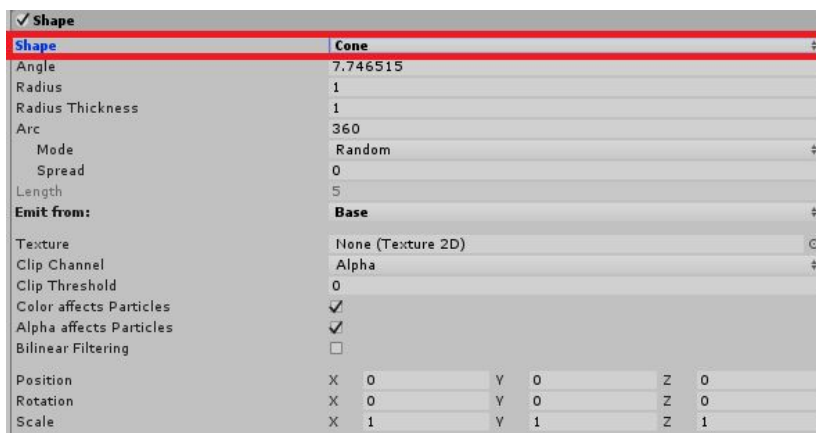
F68 - Inspector de Material



F69 - Inspector de Material 2

### 5.9.1 Partículas de Lanzar Fuego:

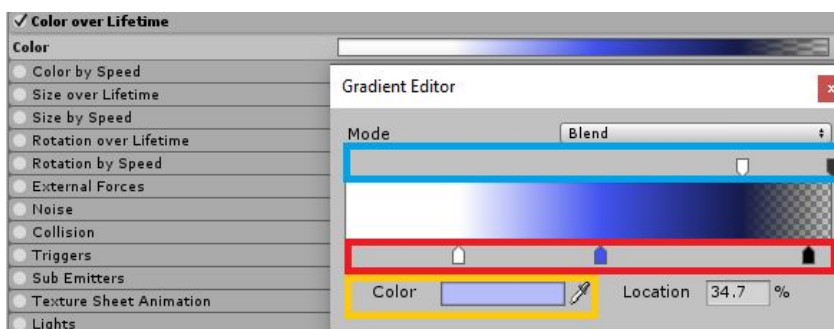
Para crear unas partículas de lanzar fuego simples como las de Morrigan, primero de todo hay que dar la forma al “ParticleSystem”. En este caso será forma de cono ya que a medida que se tira el fuego este se va expandiendo. Para conseguir este efecto hay que modificar los parámetros de “Shape” del “ParticleSystem” tal y como se muestra en la figura[F70]. Los demás valores dependen del proyecto y del resultado final que se quiera conseguir



#### F70 - Parámetro Shape

Para darles color hay que entender que el fuego en su inicio será brillante y de un color prácticamente blanco y a medida que se aleje tomará el color del

fuego, en el caso de Morrigan azul, y finalmente cambiará hacia el negro del humo. Para ello hay que modificar el parámetro “Color over lifetime” del “ParticleSystem” como se muestra en la figura[F71]. Para añadir colores al hay que pulsar en la zona del rectángulo rojo, por otro lado en el rectángulo azul podemos añadir transparencias. Y en el rectángulo naranja, con un color seleccionado, podremos modificarlo.



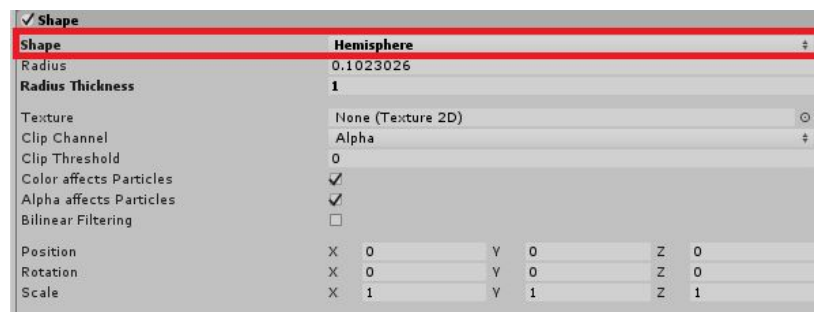
#### F71 - Parámetro Color

Esto sería lo fundamental para crear un lanzallamas sencillo. Por otro lado se pueden modificar aspectos como el tiempo de vida de las

partículas, el número de partículas que emite el “ParticleSystem”, la velocidad, el tamaño... en las distintas opciones que tiene el componente. Depende de las necesidades de cada uno adaptarse al proyecto que se está realizando.

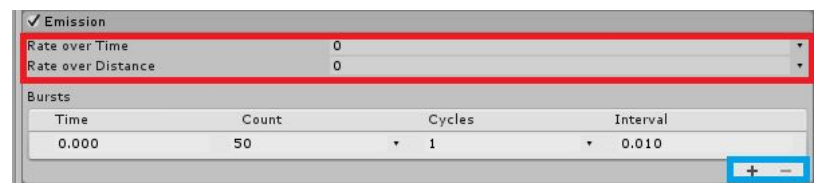
### 5.9.2 Partículas Splash Sangre:

Para crear las partículas de sangre hay que empezar modificando el parámetro “Shape” del “ParticleSystem” por una semiesfera, ya que queremos que las partículas se esparzan en todas direcciones, tal y como se muestra en la figura[F72].



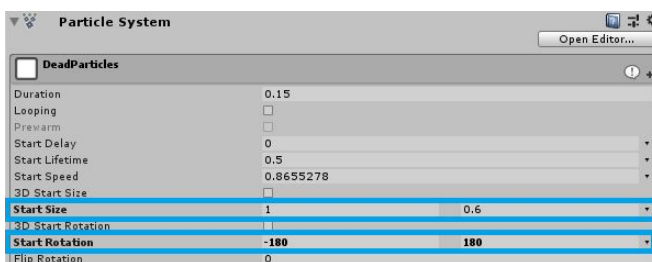
F72 - Parámetro Shape 2

Ya que la sangre crea una especie de explosión en la que se expulsan todas las partículas de golpe, tenemos que modificar el parámetro “Emission” para evitar que salgan continuamente del emisor. Para ello hay que poner las variables del



F73 - Parámetro Emission

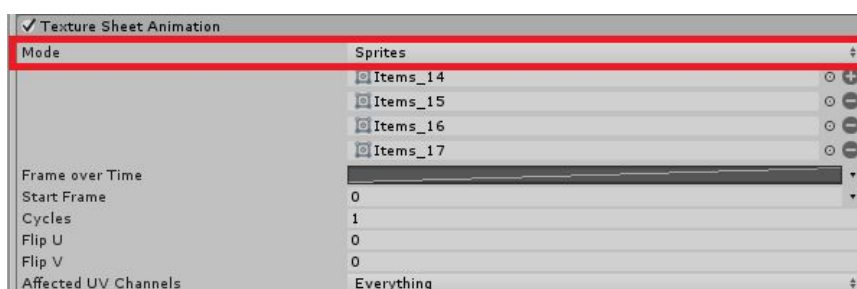
rectángulo rojo tal y como se muestra en la figura[F73]. Para añadir la explosión de partículas hay que añadir un “Burst” pulsando en símbolo “+” del rectángulo azul. Los valores del “Burst” dependen del resultado que se quiera conseguir.



F74 - Parámetro ParticleSystem

Para añadir variedad a esta explosión modificamos los valores de tamaño y rotación de manera que sean aleatorios. Para ello tal y como se muestra en la figura[F74] pulsamos en la flecha de la derecha de los rectángulos azules y seleccionamos “Random Between two Constants”

Finalmente añadimos una pequeña animación para que parezca que las partículas de sangre desaparecen pero sin modificar la transparencia. Para ello modificamos el parámetro “Texture Sheet Animation” tal y como se muestra en la figura[F75], cambiando el “Mode” a “Sprites” y añadiendo nuestra animación del spritesheet.

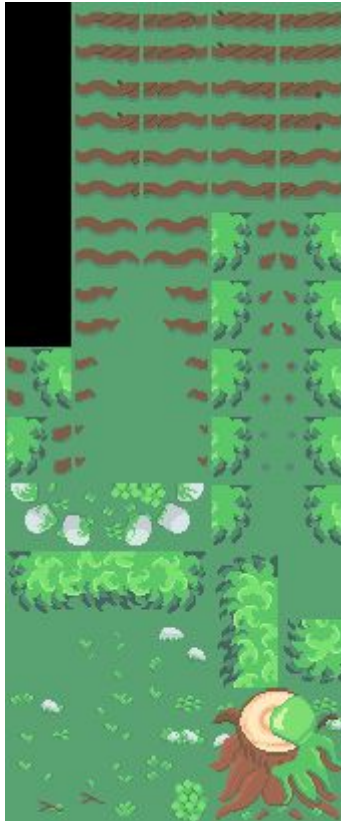


F75 - Parámetro Texture Sheet Animation

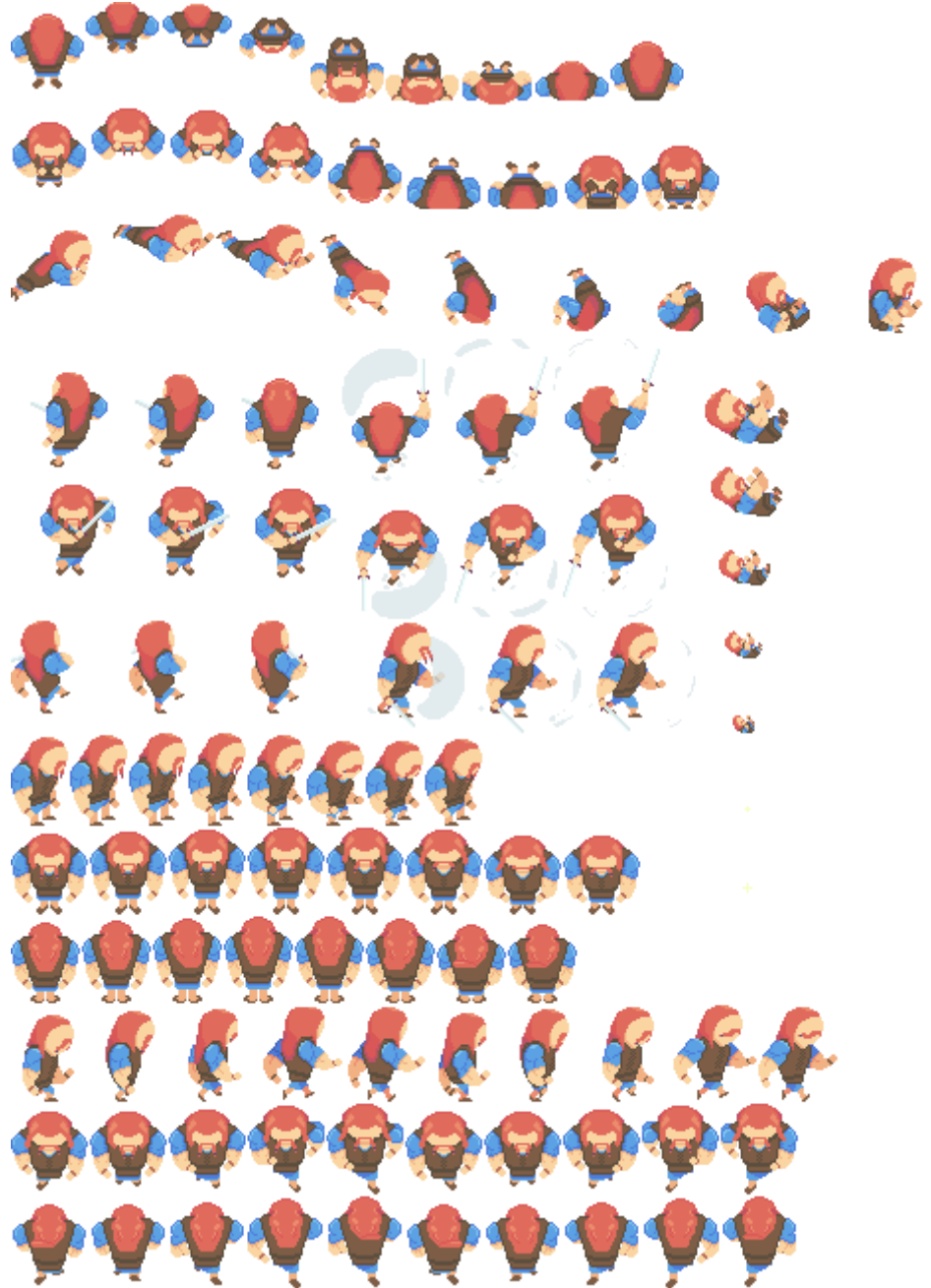


## 5.10 SpriteSheets Finales:

En este apartado están todos los spritesheets utilizados en el juego, de modo que se pueden ver mucho mejor todas las animaciones y arte que se ha desarrollado para el proyecto.



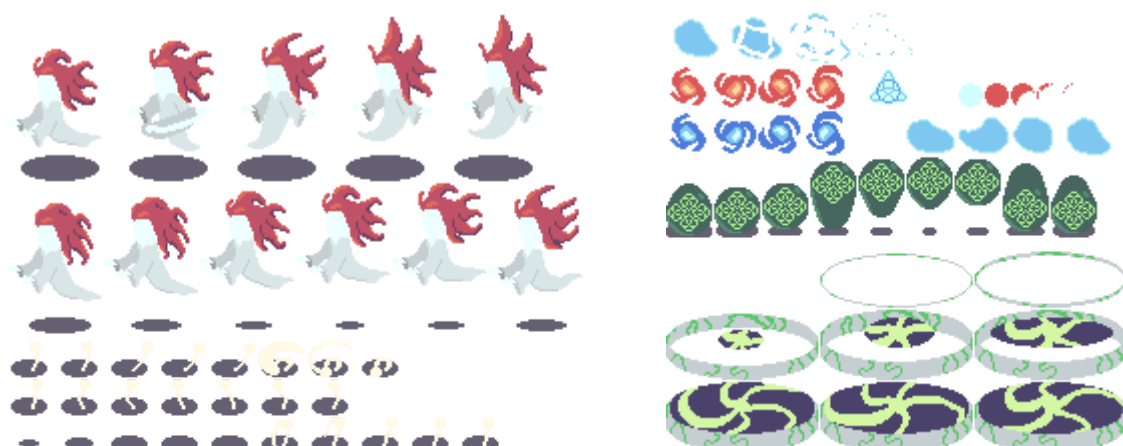
F76 - Tileset



F77 - Spritesheet Aideen







F79 - Spritesheet Objetos



F80 - Spritesheet Enemigos

### 5.11 Limitaciones durante el Desarrollo:

Una de las limitaciones más grandes que me encontré durante el desarrollo fué el algoritmo de generación de mundo, como ya he dicho en el apartado de Tileset ya que sobretodo afecta a la variabilidad del entorno y a su riqueza visual.

Por ejemplo estaba pensado añadir zonas de distintos colores, con vegetación más oscura o más clara, árboles en las salas, pequeños caminos de tierra, distintos tipos de paredes, además de agujeros y la correspondiente animación de caída del personaje principal figura[F82].

Esto ha supuesto que el apartado más flojo del juego en mi opinión sea el escenario, que muchas veces cae en la monotonía.

Por otro lado la mitología celta, aunque rica en folklore y leyendas, es demasiado extensa y abarca periodos de tiempo muy distintos, desde la edad de hierro hasta después de la cristiandad además de que los celtas nunca tuvieron una unificación como pueblo, sino que más bien eran tribus que compartían rasgos entre ellas.

Esto se refleja en su folklore que puede llegar a mezclar vampiros y demonios con seres de la naturaleza que cambian de forma. Además de que esta mitología no era común entre todos los pueblos celtas de modo que tenemos divinidades que comparten espacios y generan un panteón de deidades muy complejo para un proyecto de este tipo. A parte de que todas se transmitían oralmente por parte de los druidas y fueron transcritas muy pobremente por parte de la cristiandad. Además la influencia del imperio romano afectó a como eran estas divinidades originalmente asimilando algunos de estos dioses características de su homólogo romano.



F81 - Tiles Pared no Usadas



F82 - Animación de Caída

## 6. Bibliografia

1. [Animación Esqueletal y Animación Tradicional](#)
  - <https://www.techiediaries.com/skeletal-vs-sprite-sheet-animations-tutorial/>
2. [Spine](#)
  - <http://es.esotericsoftware.com/>
3. [Aseprite](#)
  - <https://www.aseprite.org/>
4. [PixelEdit](#)
  - <https://pyxeledit.com/>
5. [Github](#)
  - <https://github.com/>
6. [Hack'n Plan](#)
  - <https://hacknplan.com/>
7. [Gant](#)
  - <https://www.teamgantt.com/>
8. [Tipos de Animación](#)
  - <http://ezinearticles.com/?Pros-and-Cons-of-The-4-Types-of-2D-Animation&id=8868962>
9. [Simbología Celta 1](#)
  - [https://www.ancient-symbols.com/celtic\\_symbols.html](https://www.ancient-symbols.com/celtic_symbols.html)
10. [Simbología Celta 2](#)
  - <https://mythologian.net/celtic-symbols-meanings/>
11. [Mitología Celta Morrigan](#)
  - <https://www.learnreligions.com/the-morrighan-of-ireland-2561971>